

(12)特許協力条約に基づいて公開された国際出願

(19) 世界知的所有権機関
国際事務局(43) 国際公開日
2002 年 10 月 17 日 (17.10.2002)

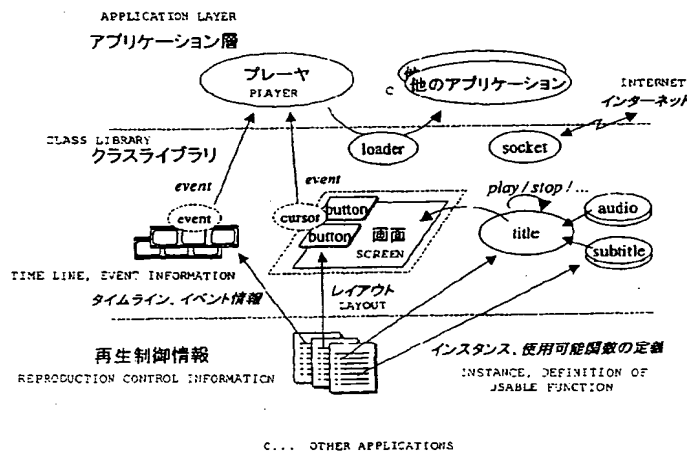
PCT

(10) 国際公開番号
WO 02/082810 A1

- (51) 国際特許分類: H04N 5/91, 5/92, 5/76, G11B 20/10, 27/00
- (21) 国際出願番号: PCT/JP02/03152
- (22) 国際出願日: 2002 年 3 月 29 日 (29.03.2002)
- (25) 国際出願の言語: 日本語
- (26) 国際公開の言語: 日本語
- (30) 優先権データ: 特願2001-103375 2001 年 4 月 2 日 (02.04.2001) JP
- (71) 出願人: 松下電器産業株式会社 (MATSUSHITA ELECTRIC INDUSTRIAL CO., LTD.) [JP/JP]; 〒571-8501 大阪府 門真市 大字門真 1 0 0 6 番地 Osaka (JP).
- (72) 発明者: 岡田 智之 (OKADA, Tomoyuki); EH12 6AE ロジアン エディンバラ ミューレイフィールド、
- (74) 代理人: 青山 森, 外 (AOYAMA, Tamotsu et al.); 〒540-0001 大阪府 大阪市中央区 城見 1 丁目 3 番 7 号 IMP ビル 青山特許事務所 Osaka (JP).
- (81) 指定国 (国内): CN, KR.
- (84) 指定国 (広域): ヨーロッパ特許 (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR).
- 添付公開書類:
— 国際調査報告書
- 2 文字コード及び他の略語については、定期発行される各 PCT ガゼットの巻頭に掲載されている「コードと略語のガイダンスノート」を参照。

(54) Title: VIDEO REPRODUCTION APPARATUS FOR DIGITAL VIDEO CONTENT, VIDEO REPRODUCTION METHOD, VIDEO REPRODUCTION PROGRAM, AND PACKAGE MEDIUM

(54) 発明の名称: デジタル映像コンテンツの映像再生装置、映像再生方法、映像再生プログラム、パッケージメディア



(57) Abstract: A video reproduction apparatus reproduces a package medium (PM) supplied from outside. The PM includes a video content storing video data and video data reproduction control information in a predetermined format and extended application software using the content. The apparatus includes as software held in the internal memory so as to be executed, an operating system (OS) selected from a plurality of OSs, middleware absorbing a difference of functions of OSs, and a player application (PA) operating on the middleware and reproducing a video content. The middleware has a class library (CL) and the PA uniformly reproduces the video content in accordance with a predetermined format via a tool contained in the (CL). Moreover, the extended application is executed via a tool contained in the (CL) of the middleware by using the video content.

/続葉有/



(57) 要約:

本発明の映像再生装置は、外部から供給されるパッケージメディア（PM）を再生する。このPMは、映像データと、映像データの再生制御情報とを所定のフォーマットで格納する映像コンテンツと、該コンテンツを使用する拡張アプリケーションソフトとを含む。該装置は、内部メモリに予め保持され実行されるソフトとして、複数種類のオペレーティングシステム（OS）から選ばれた1のOSと、各種OSの機能の差異を吸収するミドルウェアと、ミドルウェア上で動作し、映像コンテンツを再生するプレーヤアプリケーション（PA）とを備える。ミドルウェアはクラスライブラリ（CL）を有し、PAはこのCLに含まれるツールを介して、映像コンテンツを所定のフォーマットに従って統一的に再生する。また、拡張アプリケーションは、映像コンテンツを用いて、ミドルウェアのCLに含まれるツールを介して実行される。

明 細 書

デジタル映像コンテンツの映像再生装置、映像再生方法、映像再生プログラム、
パッケージメディア

5 技術分野

本発明は、映画などのデジタル映像コンテンツを記録したパッケージメディアと、その映像再生装置、映像再生方法、及び映像再生プログラムに関する。特に、DVDを置きかえる技術であるEパッケージに関する。

背景技術

10 (パッケージビジネス)

最初にパッケージビジネスの推移について説明する。

図1は、現在および今後のパッケージビジネスの流通形態を示している。図1に示すように、パッケージビジネスの流通とは、コンテンツプロバイダが所持する映画などのコンテンツを、いかにしてユーザへ提供するかである。

15 近年、映画などのコンテンツは、DVDによって、コンテンツプロバイダからユーザに提供される。

旧来のビデオカセットによる流通に比べて、DVDではスタンプによる製造コストの削減、小スペースによる輸送コストの削減、店舗における展示スペースの削減など流通ビジネスの効率を大きく向上させることができた。

20 DVDには、高画質、高音質、ランダムアクセス機能、マルチアングル等のインタラクティブ機能などの付加価値が加わり、ビデオカセットに比べ大きなメリットがある。

次にコンテンツの価値について説明する。

25 図2は、コンテンツの価値を示す概念図である。従来からのビデオカセットは、テープ上にリニアにタイトルを記録していた。即ち、映画館での上映と同じく、映画そのものを供給する媒体であり、それ以上の価値を有していなかった。

DVDは、映画自身が持つ価値の他にも、マルチアングルやマルチストーリーといったインタラクティブ機能や、メニューからのタイトル選択とランダムアクセス、音声や字幕のマルチリンガル対応などの付加価値を有する。

コンテンツの価値は、様々な要因によって目減りをしていく。例えば、音楽には「流行」がある。音楽コンテンツの多くは、時間と共に、流行が変わりその価値は著しく目減りする。映画もまた同様にその傾向がある。

5 一方で、映画には「ストーリー」が含まれている。話しの展開を知りたい視聴者は続きを見る。逆に、ストーリーを知ってしまった視聴者にとって、続きを見る動機は薄れてしまう。即ち、個々人にとって、コンテンツの価値が目減りする。

同じ音楽を毎日聴く人が多いのに対して、同じ映画を毎日見る人が少ないのは上述の為である。統計的に考えれば、映画を見た人が増えるに従って、そのコンテンツの持つ市場での価値は徐々に減っていく。

10 図3は、時間軸におけるコンテンツの価値と、それに合わせた映画ビジネスを示した図である。横軸が時間を示し、縦軸がコンテンツの価値を示している。

映画では、タイムシフトと呼ばれる独特のビジネスモデルを持っている。映画は最初に映画館で上映され、その後、DVDなどのパッケージソフトとして個人向けに販売が行われる。続いて、衛星放送やケーブルテレビなどを使って、P a y
15 y P e r V i e wと呼ばれる有料視聴に供せられ、最後に地上波局による無料放送に供せられる。地上波放送では、もちろん個々人は無料視聴できるが、これは、企業スポンサーによる広告収入によって支えられている。

(DVDの例)

ここで、DVDを例に従来のパッケージビジネスを支える技術を説明する。特に断りが無い限り本文で言うDVDはDVD-ROM、即ち再生専用ディスクを
20 指し、DVD-RAM等の録再可能なディスクを指してはいない。

図4は、DVDに記録されているデータの構造を示す。

DVDディスクには、約4.7GB（ギガバイト）の記録領域があり、DVD
ドライブのサーボを安定回転させるリードインと呼ばれる領域に続いて、「0」
25 または「1」の2値を記録する論理アドレス空間が続き、最後にディスク記録領域の終了を示すリードアウト領域が続いている。

論理アドレス空間には、まずファイルシステム領域があり、続いてAVデータや映画のシナリオを記述したナビゲーションデータが記録されている。

ファイルシステムとは、データをファイルとディレクトリ（フォルダ）として

管理するシステムであり、DVDディスク上に記録されているAVデータやナビゲーションデータは全てファイルシステムを通してそれぞれのディレクトリ、ファイルとして扱うことができる。

図4に示すように、DVDディスク上では、「VIDEO_TS」と呼ばれる、
5 DVDビデオタイトルを格納するディレクトリがルートディレクトリ直下に置かれている。このディレクトリの中には、シナリオ管理やインタラクティビティを実現するナビゲーション情報を記録した「VIDEO_TS.IFO」、「VTS_01_0.IFO」などのファイルと、AVデータを記録した「VTS_01_0.VOB」のファイルとが置かれている。

10 AVデータとしては、ISO/IEC13818 (MPEG) に準拠したストリームが記録されている。DVDでは、MPEGストリーム1本はVOBとよばれ、複数のVOBが拡張子「.VOB」を有するファイルに記録されている。複数のVOBが順に1つのVOBファイルに記録されていくが、VOBファイルが1GBを超えた場合には、1GBを境界に複数のVOBファイルに分割されて記
15 録されていく。

ナビゲーション情報は、「VMGI」と呼ばれるディスク全体を管理する情報と、「VTSI」と呼ばれるタイトル個々に関する情報とに大別される。「VTSI」の中には、「VOB (MPEGストリーム)」の一部または、全部を一つの再生ユニットとする「Cell」を有する「PGC情報」が含まれる。「Cell」は、再生シーケンスを定義する。ここで重要なのは、「Cell」が「VOB」の一部または全部を示すために用いているのが、論理アドレス空間を基準としたアドレス情報であることである。
20

例えば、コンピュータ上のHDD (ハードディスクドライブ) の場合は、ファイルが記録、編集、消去が繰り返し行われるため、同じファイルであっても、HDD上では常に同じ場所に記録されている保証はない。ファイルシステムの最大の特徴は、ファイルがHDD上のどこに記録されていてもアプリケーションからは同じようにファイルが扱えることである。
25

DVDは、AVとPCの融合を図るため、ファイルシステムを載せながらも、論理アドレスを意識したデータ構造を取っている。民生用AV機器の性能はPC

に遠く及ばない。特に、DVD登場の当時の性能はファイルシステムを載せることすら危ぶまれていた。しかし、DVDは民生機器だけでなく、PCでの使用も大きく期待されていた。事実、最近のPC製品でDVD再生機能搭載機が珍しくない。

5 即ち、DVDは、民生機としての現実的な性能と、PCからのアクセスとの両方が望まれていた。このため、PCではファイルシステムを通してデータにアクセスでき、一方、ファイルシステムの機能を持たない民生用AV機器では、論理アドレススペースでデータにアクセスが出来るように、DVDは設計された。

10 こうして、DVDは民生機器、PC双方の幅広い支持を得ることが可能になった。

(DVD規格の課題)

図1で現在および今後のパッケージビジネスの流通形態を説明している。図1に示すように、インターネットの爆発的な普及や、デジタル放送の実用化によって、パッケージの流通は物理ディスクを用いた方法だけではない。

15 既に、幾つかのコンテンツは、インターネットでストリーム配信されている。また、近年、一時記憶媒体としてハードディスクドライブ(HDD)を搭載したセットトップボックス(STB)が現れている。このハードディスクドライブに、デジタル放送を蓄積し、後から視聴できる。このように、コンテンツビジネスを行う上での環境は劇的に変化している。

20 また、映画コンテンツの流通も、DVD等の物理媒体による流通から、デジタル放送波やインターネットを使用した電子流通へとシフトしていくと考えられる。

図5は家庭内におけるAV機器の構成を示す図である。

25 AV機器を取り囲む環境は、インターネット、デジタル放送によって大きく変わっている。例えば、AV機器のインターネット接続や、デジタル放送を受信するセットトップボックス(STB)とレコーダやテレビとの接続等、機器間での接続、即ち、家庭内ネットワークが必須になっている。

特にデジタル放送によるコンテンツ配信は、インターネットのようなユーザからの要求に応じるプル型ではなく、一方的にデータを送りつけるプッシュ型で

ある。即ち、一方的に全ての視聴者に対してコンテンツを送り届ける。この場合コンテンツの著作権を防ぐシステムが必要になる。著作権保護のシステムは、暗号化技術やそのシステム技術であるDRM（デジタル・ライト・マネージメント）によって実用化されつつある。

- 5 次に必要となる技術は、コンテンツの持つ価値を管理する技術である。例えば、図2に示したように、既存のDVDより更に進んだ付加価値と、図3で示すタイムシフトモデルのように、時期や流通状況に合わせたコンテンツの価値管理方式が必要になってくる。既存のDVDは、ディスクの売り切り（セルスルー）を基本としているため、新たな付加価値の追加や、管理などが出来る構造になっては
- 10 いない。

（コンテンツ流通の課題）

コンテンツ流通の課題として、デジタル放送システムの乱立が挙げられる。

- 日本国内では、CSデジタル放送、BSデジタル放送が実用化されており、更にCS110°と呼ばれる新たな衛星放送と地上波デジタル放送とが始ま
- 15 ろうとしている。また、欧州では、各国毎に異なるデジタル放送システムがあるが、各国のシステムはDVB（Digital Video Broadcasting）システムとして統一方向に向かっている。ただし、このDVBシステムは日本のシステムとは異なる。北米でもATSと呼ばれる独自方式が検討されている。

- デジタル放送では、現在のアナログ放送のNTSC、PAL方式等よりも更に複雑に地域毎にシステムが異なってくる。
- 20

このため、映画のような全世界を対象としたコンテンツの場合、地域毎でのオーサリングが必要となり、製作コストの高騰が予想される。

- その一つの解決方法として、DVDと同等のコンテンツを電子配信する全世界統一の電子配信用パッケージが考えられる。しかし、この電子配信用パッケージをそのまま有料放送や無料地上波放送と置き換えてしまうと、DVDと全く同等
- 25 のコンテンツを無料地上波放送でも楽しめるため、ユーザのDVD購買欲を低下させ、DVDビジネスを崩壊させる危険性がある。

そのため、コンテンツの配信時期に応じて新たな付加価値をつける技術と、例えば、ユーザに応じてコンテンツの再生使用制限を設ける等の付加価値を管理す

る技術とを必要としている。

そこで、本発明の目的は、上述の課題「コンテンツに付加価値をつける」、
「時期や流通形態に合わせたコンテンツの価値管理」を解決することである。具
体的には、ネットワーク時代に対応した新しいコンテンツビジネスを築く技術で
あるEパッケージを提供することである。

発明の開示

(解決方法)

本発明に係る映像再生装置は、外部から供給されるパッケージメディアを再生
する映像再生装置であって、

10 前記パッケージメディアは、

映像データと、前記映像データの再生を制御する再生制御情報とを所定のデ
ータフォーマットで格納する映像コンテンツと、

前記映像コンテンツを使用する拡張アプリケーションソフトウェアとを含み、
前記映像再生装置は、

15 内部メモリに予め保持され実行されるソフトウェアとして、

複数の種類のオペレーティングシステムから選ばれた一つのオペレーテ
ィングシステムと、

前記オペレーティングシステムの種類に対応した機能の差異を吸収する
ミドルウェアと、

20 前記ミドルウェア上で動作し、前記映像コンテンツを再生するプレーヤ
アプリケーションソフトウェアとを備え、

前記ミドルウェアは、

前記プレーヤアプリケーションソフトウェアが前記パッケージメディア
を再生したり前記拡張アプリケーションソフトウェアを実行するために用いるツ
ールを含むクラスライブラリを有し、

25 前記プレーヤアプリケーションソフトウェアは、前記ミドルウェアの前記ク
ラスライブラリに含まれる前記ツールを介して、前記パッケージメディアの前記
映像コンテンツを、前記所定のフォーマットに従って統一的に再生し、

前記拡張アプリケーションソフトウェアは、同一のパッケージメディアに含

まれる前記映像コンテンツを用いて、前記ミドルウェアの前記クラスライブラリに含まれる前記ツールを介して実行される。

この映像再生装置は、Eパッケージの映像コンテンツを再生する。上記オペレーティングシステムは、例えば、マイクロソフト社によるウインドウズ（登録商標）、アップル社によるMac OS（登録商標）、又はフリーウェアのLinux等を用いることができる。なお、オペレーティングシステムは上記のものに限られず、各メーカーによるOSを含む。また、上記ミドルウェアとしては、例えば、Javaを用いることができる。このミドルウェアによってオペレーティングシステムの種類が異なる場合にも機能の差異を吸収できる。また、上記プレーヤアプリケーションソフトウェアは、パッケージメディアの映像コンテンツを再生する。さらに、拡張アプリケーションソフトウェアとしては、例えば、上記パッケージメディアの映像コンテンツを利用するゲームアプリケーション等である。なお、上記プレーヤアプリケーションソフトウェア及び拡張アプリケーションソフトウェアは、上記ミドルウェア上で動作する。ミドルウェアは、上記アプリケーションソフトウェアが映像コンテンツを再生したり、実行する際に使用されるツールを含むクラスライブラリを有する。このクラスライブラリに含まれるツールとは、例えば、種々の機能を実現するクラスやそのメンバ関数を意味する。また、さらに、この映像再生システムは、ネットワークを介して配信されたソフトウェアを実行することによって構成することもできる。

また、この映像再生装置は、再生ステータス情報を管理しており、前記パッケージメディアの前記再生制御情報は前記再生ステータス情報に対応する再生制限情報を含み、前記再生制御情報を解析し、前記再生制御情報内の再生制限情報と前記再生ステータス情報の比較により、前記拡張アプリケーションソフトウェアが前記ミドルウェアの前記クラスライブラリに含まれる前記ツールのいずれかを無効に設定する。

本発明に係る映像再生方法は、外部から供給されるパッケージメディアを映像再生装置で再生する映像再生方法であって、

前記パッケージメディアは、

映像データと、前記映像データの再生を制御する再生制御情報とを所定のデ

ータフォーマットで格納する映像コンテンツと、

前記映像コンテンツを使用する拡張アプリケーションソフトウェアとを含み、
前記映像再生方法は、

5 複数の種類のオペレーティングシステムから選ばれた一つのオペレーティングシステムを前記映像再生装置の内部メモリに読み込み、起動するステップと、
該オペレーティングシステムの種類に対応した機能の差異を吸収するミドルウェアであって、該ミドルウェア上で動作するアプリケーションソフトウェアが前記パッケージメディアを再生したり、実行するために用いるツールを含むクラスライブラリを有するミドルウェアを前記映像再生装置の前記内部メモリに読み込み、起動するステップと、

10 前記ミドルウェア上で動作し、前記映像コンテンツを再生するプレーヤアプリケーションソフトウェアを前記映像再生装置の内部メモリに読み込み、起動するステップと、

15 前記ミドルウェア上で動作し、前記映像コンテンツを使用する拡張アプリケーションソフトウェアを前記映像再生装置の内部メモリに読み込み、起動するステップと、

20 前記プレーヤアプリケーションソフトウェアによって、前記ミドルウェアの前記クラスライブラリに含まれるツールを介して、前記パッケージメディアの前記映像コンテンツを、前記所定のフォーマットに従って統一的に再生するステップと、

前記拡張アプリケーションソフトウェアによって、前記映像コンテンツを用いて、前記ミドルウェアの前記クラスライブラリに含まれる前記ツールを介して実行するステップとを含む。

25 本発明に係る映像再生プログラムは、外部から供給されるパッケージメディアを再生する映像再生プログラムであって、

前記パッケージメディアは、

映像データと、前記映像データの再生を制御する再生制御情報とを所定のデータフォーマットで格納する映像コンテンツと、

前記映像コンテンツを使用する拡張アプリケーションソフトウェアとを含み、
前記映像再生プログラムは、

内部メモリに予め保持され実行されるソフトウェアとして、

5 複数の種類のオペレーティングシステムから選ばれた一つのオペレーティングシステムと、

前記オペレーティングシステムの種類に対応した機能の差異を吸収するミドルウェアと、

前記ミドルウェア上で動作し、前記映像コンテンツを再生するプレーヤアプリケーションソフトウェアとを備え、

10 前記ミドルウェアは、

前記プレーヤアプリケーションソフトウェアが前記パッケージメディアを再生したり前記拡張アプリケーションソフトウェアを実行するために用いるツールを含むクラスライブラリを有し、

15 前記プレーヤアプリケーションソフトウェアは、前記ミドルウェアの前記クラスライブラリに含まれる前記ツールを介して、前記パッケージメディアの前記映像コンテンツを、前記所定のフォーマットに従って統一的に再生し、

前記拡張アプリケーションソフトウェアは、同一のパッケージメディアに含まれる前記映像コンテンツを用いて、前記ミドルウェアの前記クラスライブラリに含まれる前記ツールを介して実行される、映像再生プログラム。

20 また、本発明に係るコンピュータ読み取り可能な記録媒体は、前記映像再生プログラムを格納する。

本発明に係るパッケージメディアは、
映像再生装置に外部から供給され、該映像再生装置で再生されるパッケージメディアであって、

25 前記パッケージメディアは、

映像データと、前記映像データの再生を制御する再生制御情報とを所定のデータフォーマットで格納する映像コンテンツと、

前記映像コンテンツを使用する拡張アプリケーションソフトウェアとを含み、
前記映像再生装置は、

内部メモリに予め保持され実行されるソフトウェアとして、

複数の種類のオペレーティングシステムから選ばれた一つのオペレーティングシステムと、

5 前記オペレーティングシステムの種類に対応した機能の差異を吸収するミドルウェアと、

前記ミドルウェア上で動作し、前記映像コンテンツを再生するプレーヤーアプリケーションソフトウェアとを備え、

前記ミドルウェアは、

10 前記プレーヤーアプリケーションソフトウェアが前記パッケージメディアを再生したり前記拡張アプリケーションソフトウェアを実行するために用いるツールを含むクラスライブラリを有し、

前記プレーヤーアプリケーションソフトウェアは、前記ミドルウェアの前記クラスライブラリに含まれる前記ツールを介して、前記パッケージメディアの前記映像コンテンツを、前記所定のフォーマットに従って統一的に再生し、

15 前記拡張アプリケーションソフトウェアは、同一のパッケージメディアに含まれる前記映像コンテンツを用いて、前記ミドルウェアの前記クラスライブラリに含まれる前記ツールを介して実行される。

このパッケージメディアは、高付加価値を有するEパッケージである。即ち、このパッケージメディアの映像コンテンツをプレーヤーアプリケーションソフトウェアで再生するだけでなく、上記映像コンテンツを利用するゲームアプリケーションソフトウェア等に関連させて実行することができる。また、このパッケージメディアでは、再生制御情報には、映像データの再生順序を規定するシナリオ情報を含んでいてもよい。さらに、再生制御情報は、映像コンテンツの再生やゲームアプリケーションの利用を制限するレベルを規定する再生レベル情報を含んで

20

25

いてもよい。

図面の簡単な説明

本発明の種々の対象、特徴及び利点は、添付の図面を参照しつつ以下で説明される好ましい実施の形態により明らかにされるであろう。

図1は、パッケージビジネスを示す概念図である。

図 2 は、コンテンツの価値を示す概念図である。

図 3 は、映画におけるタイムシフトビジネスを示す概念図である。

図 4 は、DVD 規格の構成図である。

図 5 は、家庭内における AV 機器の構成図である。

5 図 6 は、映画タイトル間でのリンクの概念図である。

図 7 は、新たな価値を示す概念図である。

図 8 は、E パッケージのレベルを示す概念図である。

図 9 は、様々な規格の構成図である。

図 10 は、ミドルウェア上でのプレーヤモデル構成図である。

10 図 11 は、アプリケーションの例「プレーヤ」の概念図である。

図 12 は、アプリケーションの例「ゲーム」の概念図である。

図 13 は、アプリケーションの例「映画間リンク」の概念図である。

図 14 は、E パッケージ規格の構成図である。

図 15 は、ディレクトリ、ファイルの構成図である。

15 図 16 は、データ構造図「パッケージ情報」である。

図 17 は、データ構造図「メニュー情報」である。

図 18 は、データ構造図「タイトル情報」である。

図 19 は、データ構造図「ストリーム情報」である。

図 20 は、データ構造図「字幕ストリーム」である。

20 図 21 は、ストリーム構造図である。

図 22 は、映像再生装置構成図である。

図 23 は、ソフトウェア構成図である。

図 24 は、クラス一覧を示す図である。

図 25 は、Package クラスの処理フローである。

25 図 26 は、Title クラスの処理フローである。

図 27 は、Menu クラスの処理フローである。

図 28 は、Audio クラスの処理フローである。

図 29 は、Event クラスおよび Link クラスの処理フローである。

図 30 は、プレーヤ再生フローである。

図 3 1 は、メニュー例の図である。

図 3 2 は、タイトル再生中の動作例の図である。

図 3 3 は、enable Event 関数の処理フローである。

図 3 4 は、Cursor クラスの処理フローである。

5 図 3 5 は、Status クラスの処理フローである。

図 3 6 は、Canvas クラスの処理フローである。

図 3 7 は、ゲームアプリケーション再生フローである。

図 3 8 は、ステータス更新の概念図である。

図 3 9 は、アップデートアプリケーション処理フローである。

10 発明を実施するための最良の形態

以下、図面を参照して本発明に係る好ましい実施形態について説明する。なお、図面において、同一の符号は同様のものを示す。

(新たなビジネスの形態)

15 図 2、図 3 で説明したように、映画コンテンツの価値は時間と共に減少をしていく。更に、世界的に電子流通を進めていくためには、既存のビジネスモデルを変えていかなければならない。

本発明の実施の形態に係るデジタル映像コンテンツを含むパッケージメディア（以下、「E パッケージ」という）は、図 2 で示すように映画そのものに匹敵するアプリケーションを付加価値として導入する。これにより、パッケージの価値を高めることができる。また、図 5 で示すように、アプリケーションによって導かれる価値をコントロールし、パッケージにレベルを持たせ、同じタイトルであつても差別化を図れるようにする。

20 例えば、図 8 で示しているように、全てのアプリケーションを使用可能としたパッケージを「フルパッケージ」、一部のアプリケーションの使用を制限したパッケージを「制限付きパッケージ」、映画コンテンツだけの視聴を可能としたパッケージを「無料パッケージ」としてレベルを設けて価値をコントロールする。

25 図 3 で示しているパッケージビジネスにおいて、既存の DVD に代わってフルパッケージを流通させ、有料放送に代わって制限付きパッケージを、無料放送に代わって無料パッケージをそれぞれ流通させることができる。

本実施の形態では、3通りのレベルでしか説明を行わないが、よりパッケージレベルを細分化して、細かく流通ビジネスを展開することもできる。

(様々な規格の構成)

5 図9を用いて、代表的なメディアであるCD、DVD、DVB-MHPおよびEパッケージそれぞれの規格の構成、概要について説明する。なお、VHSは、物理的な特性と電気信号とから構成され、データ構造を有する他の規格とは大きく異なるので、VHSについての説明は省略する。

10 CDは、サンプリング周波数44.1kHzで標本化されたデータと、TOC (Table Of Contents) と呼ばれる各トラック(曲)へのインデックス情報とから構成されている。CD再生機は、TOCを読み込み、ユーザからのリクエスト、例えば「トラック3の再生」など、を受けて、対応する曲のデータを読み出し、DA変換を行って再生する。

15 図示はしていないが、CDを改良したVideo CDには、AVストリームと、そのインデックスとなるPSD (Programmable Sequence Descriptor) とが記録されている。Video CD再生機は、ユーザの操作に従って、対応するAVストリームの読み出しとデコードを行って再生する。

CDとVideo CDのどちらも、データ構造が規格とされ、再生機器は規格に従い、データ構造を解釈し実行している。

20 DVDには仮想マシンという概念が導入された。これはCPUと同じように、演算処理機能とレジスタ(専用メモリ)を有する構成である。データ構造として記述されているシナリオデータに対して、ユーザの操作やレジスタの値によって、プレーヤの動作が異なってくる。

25 簡単な例を説明すれば、映画の視聴者が「大人」であるか、「18歳未満の子供」であるかによってストーリーを分けることができる。これは、パレンタルロックと呼ばれる機能である。これによって、映画の中にある性的シーン、暴力シーンを子供に見せないようにカットする。他にもユーザが男性であるか、女性であるかによってストーリーやアングルを変えることもできる。

DVDでは、静的なデータ構造の他に、仮想マシンとしてプレーヤ(映像再生

装置ともいう)の動作モデルを規格として定義している。これによって、プレーヤメーカー間で異なるハードウェアやソフトウェアのプラットフォームや、プレーヤアプリケーションのソフトウェアインプリメンテーションの差異を吸収し、プレーヤ間での互換性を実現する。

5 次に、DVB-MHP (Digital Video Broadcasting Multimedia Home Platform) について説明する。DVB-MHPは、欧州で標準化が進んでいる次世代のデジタル放送規格である。この規格の最大の特徴は、Javaと呼ばれるミドルウェアを用いていることである。

10 Javaは、プラットフォーム間での互換性を改善するためにSun Microsystems社が提唱したミドルウェアである。Javaを搭載したコンピュータ、機器上では、全てのJavaアプリケーションが動作することができ、プラットフォームの垣根を越えてアプリケーションの使用範囲を大きく広げられることが最大の特徴である。

15 国内でも、NTT DoCoMoのiモードや、AV機器間のネットワークを目指しているHaviなどで、Javaが採用の動きがある。

DVB-MHPでは、Javaの導入とともに、DVB-MHPに特化した、即ち、テレビ放送の映像番組やデータ放送番組を処理するオブジェクトクラスとそのインターフェースを定義している。

20 DVB-MHPは、従来の規格と比較すると、静的なデータ構造を決めているのではなく、ミドルウェア上のインターフェースを規格として定義している点で大きく異なっている。

このため、アプリケーションには、コンピュータプログラムとして作ることが出来る全てのものが使用できる。その反面、アプリケーションを作るシステムが何も設けられていない。そのため、使用されるアプリケーションは、従来のコン
25 テンツビジネスと比較すると、音楽や映画などのAV系よりもコンピュータゲームの方に近い。

本発明のEパッケージでは、DVB-MHPと同じようにミドルウェアを載せ、プレーヤ上で様々なアプリケーションが動作できるようにしている。しかし、映

画という最大のコンテンツを効率良く作るには、従来型の静的なデータ構造と、仮想マシンのようなプレーヤ動作モデルとを有するほうが便利である。

そこで、本発明のEパッケージは、映画コンテンツ用に静的なデータ構造とプレーヤ動作モデルを定義する。また、このEパッケージは、映画コンテンツの価値を高めるアプリケーションに対してのインターフェースを提供する。

(プレーヤモデル)

図10はミドルウェア上でのプレーヤモデル概念図である。

ここで使用するミドルウェアはJavaのようなオブジェクト思考プログラミング言語である。オブジェクト思考プログラミング言語およびその基本クラスそのものについては、その多くが書籍またはインターネット上のホームページにおいて公開されているため、その詳細、特にクラスライブラリ内部での処理は省略する。

Eパッケージのミドルウェアでは、タイトルや言語設定など様々な機能がクラスおよびそのメンバ関数として定義される。各クラスは実行時にそのインスタンスが生成され、プレーヤアプリケーション等のアプリケーションからアクセスされる。

ここで使用するクラスを簡単に説明する。図10中の楕円は各クラスのインスタンスを示している。

「Titleクラス」は、各映画タイトルに相当するEパッケージ専用のクラスである。このクラスは、チャプタなどのシナリオ情報、AVデータのアドレス情報、アプリケーションに対して提供するインターフェース情報等を有する。

これらの情報は全て再生制御情報ファイル(図中の下段)に記述されている。再生制御情報によって記述された属性は、そのままオブジェクトの属性となる。例えば、Titleインスタンスのレベル属性は、再生制御情報Titleの属性levelによって指定される。詳細は後述する。

また、Titleクラスは再生制御用のメンバ関数(Methods)を有する。

例えば、タイトルの再生は「Play()」関数をコールすることで行われ、再生停止は、「Stop()」関数をコールすることで行われる。

これらのメンバ関数 (Methods) も、再生制御情報によって、その機能が制御される。例えば、TitleインスタンスのSetRate (特殊再生関数) は、再生制御情報の<SETRATE level="">によって、その関数の使用が制限される。詳細は後述する。

- 5 「Audioクラス」は、オーディオストリームに相当するクラスである。このクラスは各オーディオストリーム毎にインスタンス化される。インスタンスは、ストリームの属性や言語情報などを持つ。例えば、オーディオストリームの言語情報は、<AUDIO language="Japanese">のようにして再生制御情報で定義される。この属性値は、Audioインスタンスからメンバ関数getLang () によって取り出す事が可能である。

- 10 Eパッケージは、DVDと同様にマルチ言語に対応しており、ユーザは好みの音声ストリームの選択ができる。プレーヤアプリケーションはユーザからのリクエストを受け、対応するインスタンスをTitleクラスインスタンスへ指定 (set) する。この時、前述したように各Audioインスタンスのメンバ関数getLang () を使って、対応言語を調べ、ユーザのリクエストにあったAudioインスタンスを選び、Titleインスタンスへ指定 (set) する。

「Subtitleクラス」は、字幕ストリームに相当するクラスであり、Audioクラスとほぼ同様の機能を有している。

- 20 「Socketクラス」は、ネットワークを経由して他のプレーヤ (映像再生装置) およびサーバ等と通信するクラスである。

- 「Loaderクラス」は、他のアプリケーションなどを動的に組み込むクラスである。Loaderクラスによって動的に組み込まれるアプリケーションは、再生制御情報ファイルによって定義される。プレーヤアプリケーションを使って他のアプリケーションを再生する場合には、通常、Loaderクラスを使用する。しかし、プレーヤ機能も備えたアプリケーションが実行される場合には、必ずしもLoaderクラスを呼ぶ必要はない。

- 25 「Eventクラス」は、シナリオ内に記述されたイベントトリガを発生するクラスである。例えば、映画の途中でユーザにダイアログメッセージを表示するなどに用いることができる。

「C u r s o r クラス」はユーザのカーソル移動をアプリケーションに伝えるクラスである。リモコンによるカーソルの移動などをキャッチする。

「B u t t o n クラス」、「C a n v a s クラス」、「F r a m e クラス」はそれぞれ画面上にボタン、キャンバス、フレームを表示するクラスである。これ
5 らのクラスはインスタンスを生成して、画面上に組み込む（a d d）ことで描画が行われる。

特にC a n v a s クラスは、動画を描画するクラスである。このC a n v a s クラスのインスタンス上にT i t l e インスタンスを組み込む（a d d）ことで画面上に動画表示を行うことができる。また、T i t l e インスタンスを外す
10 （d e l e t e）ことで動画表示を終了させることができる。

「T e x t クラス」は、画面上にテキスト表示を行うクラスである。コンストラクタによって、T e x t インスタンスを生成し、C a n v a s インスタンス上にT e x t インスタンスをおく（a d d）ことで画面上に任意のテキストを描画
15 することができる。

（アプリケーションの例）

上述したプレーヤモデルの構成によって、以下に記すアプリケーションを実現
20 できる。

図11に示すのはシンプルなDVDプレーヤの例である。

図11に示す通り、DVDプレーヤアプリケーションも一つのアプリケーションとしてミドルウェアの上に実装されている。プレーヤアプリケーションは、ミ
20 ドルウェア上で提供されるクラスライブラリからインスタンスを生成し、そのメソッド関数を呼び出すことで、タイトルの再生を行う。

例えば、T i t l e クラスから作られるメニューインスタンスをC a n v a s インスタンスに設定（a d d）することで画面上にメニューを表示し、ユーザからのリクエストを受けつける。ユーザはカーソルを使い再生するタイトルを選択
25 する。

ユーザからのリクエストは、C u r s o r クラスのインスタンスを通してタイトルまたはメニューに届けられる。例えば、メニューでは、ユーザが選択したタイトルに対応するT i t l e クラスのインスタンスを取得し、C a n v a s イン

スタンスに載せ (a d d)、再生 (p l a y) を実行する。

図 1 2 に示すのはゲームアプリケーションの例である。

図 1 2 では、プレーヤアプリケーションに代わってゲームアプリケーションが
起動されている。ゲームアプリケーションは、パッケージ内のタイトルから任意
5 の画面を選びゲームの背景画面として表示している。ゲームアプリケーションは
3 D ポリゴン画像を背景画像の上に載せ、ゲームを進行していく。基本的な動作
は前述したプレーヤアプリケーションの例と同じであり、アプリケーションプロ
グラムが専用プレーヤアプリケーションでなく、ゲームアプリケーションに代わ
っている。

10 勿論、背景画面を細かく制御して、ゲームと同期させた表示を行うことも可能
である。

図 1 3 に示すのはタイトル間のリンク構造である。

前述したように、ホームサーバ上には多数の映画コンテンツが記録されている。
どの映画タイトルが実際に記録されているのかは、各家庭によってバラバラであ
15 り、図 1 3 に示すようにタイトル間に跨る構造は一意に規定することはできない。

そこで、本発明の E パッケージでは、各タイトル毎にリンク先の情報を有し、
実際に有効なリンク先だけが再生時に有効となる構成を取っている。

例えば、「T i t l e 1」は、「T i t l e 2」、「T i t l e 3」、「T i
t l e 5」、「T i t l e 6」のリンク情報を有している。一方、ホームサーバ
20 上には「T i t l e 5」が存在しない。この場合、「T i t l e 1」再生時の有
効リンクは、「T i t l e 2」、「T i t l e 3」または「T i t l e 6」にな
る。このように動的に再生可能なリンクだけを選び出すことを可能としている。

(規格の構成)

図 1 4 は規格の構成を示す図である。

25 図 1 4 に示すように E パッケージ規格は、プレーヤモデル、データ構造、A V
データの主に 3 つのパートから構成されている。

プレーヤモデルは、オブジェクト指向プログラミング言語のクラスライブラリ
として構成されており、再生制御情報に基づいてメニューやタイトルなどの機能
をそれぞれインスタンス化してアプリケーションに提供する。

データ構造は、図14に示すように、パッケージ全体を管理するパッケージ情報、メニューを記述したメニュー情報、各タイトル毎にシナリオを記述したタイトル情報、ストリーム毎に属性やアクセスするためのアドレスなどを記述したストリーム情報等から構成されている。以下詳細を説明する。

5 まず、図15を用いて、パッケージのディレクトリ、ファイル構造を説明する。

Eパッケージは、DVDと同じように単体の光ディスクとして流通する場合と、ネットワークを通じて電子的に配信、HDDに蓄積される場合とがある。ここで説明するディレクトリ（フォルダという場合もある）、ファイル構造はどちらでも共通に使われる形式である。

10 Eパッケージは、DVDと同様にファイルシステムを導入している。

Eパッケージのファイルシステムでは、「PACKAGE」ディレクトリがルートディレクトリの直下に置かれている。このディレクトリはEパッケージ専用のディレクトリであって、他のアプリケーション、例えば従来のDVDデータなどは置かれていない。「PACKAGE」ディレクトリの下に更にサブディレクトリが置かれ、それぞれが一つのパッケージに対応している。図15での「abc」、「あいう」がそれに対応する。

サブディレクトリの中に各管理情報に対応するファイルと、ストリームデータが置かれている。最初の「package.xml」は予約ファイルであって、前述の「パッケージ情報」が記録されている。その他には、メニューを記述した「menu.xml」、タイトルを記述した「title1.xml」、「title2.xml」、ストリーム情報を記録した「stream1.xml」、「stream2.xml」が置かれている。

(データ構造詳細)

図16はパッケージ情報「package.xml」の詳細を示している。

25 データの記述は前述した通り、XMLの記述方式に従い、タグ<PACKAGE>で囲まれた中にパッケージ情報が記述され、以下の情報を有している。

<GENERAL> 一般情報

バージョン情報 (version)

<ACCESS> アクセス制限情報

地域情報 (region)

Eパッケージは、この地域情報を用いて映像コンテンツへのアクセス制限を行うことにより、映像コンテンツを再生できる地域を限定できる。この地域情報に再生可能な地域を順次増やしていくか、又は地域ごとに該当する地域情報を付与することによって、映画タイトルのタイムシフト、例えば北米を皮切りに日本および欧州、アジア、中国の順にタイトルを供給していくことが可能である。地域情報 (region) は、それぞれ、“US”、“Japan”、“EU”、“Asia”、“China”などの値を持つ。

<UPDATE> 更新予告情報

日時情報 (data)

自動取得の実施フラグ (auto)

この更新予告情報には、シナリオ他、映画タイトルの自動更新予定が記述される。プレーヤ (映像再生装置) は、この情報に基づいて自動的にインターネットを通じて新しい情報への更新ができる。

<INTERNET> インターネットホームページ情報

URL (URL)

このインターネットホームページ情報は、関連する情報が記載されているインターネットのホームページアドレス情報である。ユーザからインターネットアクセスの要求があった場合に、この情報が示すアドレスへアクセスする。

また、前述の更新予告でも上記のアドレスを基に情報の取得を行う。

<MENU> メニュー情報

メニュー情報ファイル (menu)

メニュー情報は、メニュー情報ファイルを指定する。指定されたファイルの中にメニュー情報が記述される。

<TITLE_LIST> タイトルリスト

パッケージで取り扱うタイトルは<TITLE_LIST>タグに囲まれた中で<TITLE>タグによって記述される。

<TITLE> タイトル情報

タイトル番号情報 (number)

タイトル情報ファイル (file)

タイトル情報には、各タイトルへのリンク情報が記述される。個々のタイトル自身は、指定されたタイトル情報ファイル内に記述されている。

図17はメニュー情報「menu.xml」の詳細を記している。

- 5 タグ<MENU>で囲まれた中に以下に示すメニュー情報が記述されている。

<MENU_PAGE> メニューページ情報

ページ番号 (page)

背景画像情報 (image)

- 10 メニューページ情報は、メニュー画面が複数あるマルチページメニューに関する情報である。例えば表示したいタイトルが100以上と大量にあるために1枚のページだけでは表示しきれない場合には、マルチページメニューが使われる。

<TITLE> タイトル情報

横座標 (column)

縦座標 (row)

- 15 タイトル番号 (title)

オブジェクト名 (object)

タイトル名 (<TITLE>タグに囲まれて記述される)

- 20 タイトル情報タグ毎に、各タイトルの情報が記される。プレーヤアプリケーションはこの情報を基にメニュー表示を行う。また、オブジェクトで指定される部品がグラフィカル・ユーザ・インターフェースとして、画面上に表示される。この部品はミドルウェア上のクラスライブラリの機能として提供されている。

- 25 例えば、図17のように、オブジェクト (object) がボタン (button) だった場合は、ミドルウェアのグラフィックライブラリが提供するボタンオブジェクトがメニュー上に表示される。表示位置は、横座標 (column) および縦座標 (row) によって示され、また、タイトル名がボタン上に表示される。

図18はタイトル情報「title1.xml」他、の詳細を記している。

タグ<TITLE>で囲まれた中に以下に示すタイトル情報が記述されている。

<TITLE> タイトル情報

タイトル番号 (t i t l e)

レベル (l e v e l)

レベルは、本タイトルの再生レベルを示している。前述したように、Eパッケージではパッケージの再生レベルを設けることで、ユーザの購入状況に応じて再生可能レベルを分けることができる。具体的には、レベル (l e v e l) は、フルパッケージ (f u l l)、制限つきパッケージ (r e s t r i c t e d)、無料パッケージ (f r e e) のいずれかの値を持つ。一方、プレーヤの持つステータス (S t a t u s) が、「フル再生可能 (f u l l p l a y b a c k)」である場合は、全てのパッケージの再生が可能であり、ステータス (S t a t u s) が、「制限付き再生可 (r e s t r i c t e d)」である場合 (r e s t r i c t e d p l a y b a c k) は、制限つきパッケージ (r e s t r i c t e d) または無料パッケージ (f r e e) の何れかが再生できる。プレーヤ側の属性が、「無料パッケージのみ (f r e e o n l y)」の場合は、無料パッケージ (f r e e) のみ再生可能である。

ここでは、パッケージの種類を3種類に限定したが、分類数は本質的な問題では無く、2種類、4種類またはそれ以上の種類に分類し、再生制限を行うことはもちろん可能である。また、分類の仕方や名称も上述の限りに限定されるものではない。

<L I N K _ L I S T> リンクリスト

本タイトル中で発生するリンク一覧を定義する。

<L I N K>情報

識別情報 (I D)

リンク先パッケージ情報 (p a c k a g e)

リンク先タイトル情報 (t i t l e)

リンク先チャプタ情報 (c h a p t e r)

リンク先時刻情報 (t i m e)

リンク情報がリンク情報タグ毎に記される。各リンク情報は後述するタイムライン情報内などで実際に使用される。リンク情報は、タイトル起動時にリンク先の有効、無効をプレーヤが自動検出するために定義されている。

<CHAPTER_LIST> チャプタリスト

<CHAPTER> チャプタ情報

開始時刻情報 (in)

終了時刻情報 (out)

5 再生ストリーム情報 (video)

再生字幕情報 (subtitle)

チャプタはタイトル情報内でエントリされる。

<TIMELINE> タイムライン情報

10 時間軸上に展開されるイベントなどの情報は、タイムライン情報タグ内に記述される。記述される情報は以下の通りである。

<BRANCH> 分岐情報

レベル情報 (level)

メッセージ情報 (message)

識別情報 (ID)

15 有効区間開始時刻情報 (in)

有効区間終了時刻情報 (out)

分岐先タイトル (jump)

20 レベル情報 (level) は、前述した通り映像再生装置のステータス (Status) に応じて処理を行うか否かを示すフラグである。例えば、映像再生装置のステータス (Status) が「無料パッケージのみ (free only)」で、レベル情報 (level) が「フルパッケージ (full)」であった場合などには、当該分岐タグ (BRANCH) は無視される。また、識別情報 (ID) は前述したLINK情報の識別情報 (ID) に対応している。

25 プレーヤモデルはユーザからの分岐要求を受けた場合に、対応するLINK情報に記述されている場所の再生を開始する。

<MESSAGE> メッセージ情報

レベル情報 (level)

メッセージ情報 (message)

識別情報 (ID)

有効区間開始時刻情報 (in)

有効区間終了時刻情報 (out)

メッセージタグによって記述されるメッセージは、プレーヤのOSDによって
字幕表示される。

5 <TRIGGER> イベントトリガ情報

 レベル情報 (level)

 イベント情報 (event)

 識別情報 (ID)

 発生時刻情報 (time)

10 発生時刻になった時点で、アプリケーションに対してイベントを投げる。内容は
イベント情報 (event) に記され、そのままアプリケーションに渡される。

 <INTERFACE> インターフェース情報

 <PLAY> 再生機能制御タグ

 <STOP> 停止機能制御タグ

15 <SETRATE> 特再機能制御タグ

 <SETTIME> 飛び込み再生機能制御タグ

 <SETAUDIO> 音声設定機能制御タグ

 <SETSUBTITLE> 字幕設定機能制御タグ

20 インターフェース情報 (<INTERFACE>) は、上述する幾つかのプレーヤ機能制御タグを持つ。個々のタグはTitleインスタンスのメンバ関数、
play、stop、setRate、setTime、setAudio、setSubtitleにそれぞれ対応する。また、個々のタグは属性レベル (level) を持ち、パッケージのレベル (level) と同様に“full”、
“restricted”または“free”のいずれかの値を持つ。

25 例えば、レベル (level) が“full”の場合は、対応するTitleインスタンスのメンバ関数の使用が制限される。この場合、映像再生装置が持つステータス (Status) が「フル再生可能 (full playback)」の場合にのみ当該関数の使用が可能である。各関数のレベル (level) と、プレーヤアプリケーションのステータス (Status) の関係は前述したパッ

ページでのレベル (level) と同じである。

図19はストリーム情報「stream1.xml」他、の詳細を記している。
タグ<STREAM>で囲まれた中に以下に示すタイトル情報が記述されている。

5 <STREAM> ストリーム情報

 ファイル情報 (file)

 ファイル情報は再生対象のストリームファイル名を記述する。

 <ATTRIBUTE> 属性情報

10 以下に記すビデオおよびオーディオの属性情報は属性情報タグに囲まれて記述
 される。

 <VIDEO> ビデオ属性情報

 圧縮情報 (coding)

 解像度情報 (resolution)

 アスペクト比情報 (aspect)

15 <AUDIO> オーディオ属性情報

 圧縮情報 (coding)

 ビットレート情報 (bitrate)

 チャンネル数情報 (channel)

 言語情報 (language)

20 <TIMEMAP> タイムマップ情報

 タイムマップ情報には、VOBU (詳細は後述する) 毎の時間およびサイズ情報が記述されている。各VOBUとしてエントリされているユニットの再生時間長 (フレーム数) とデータサイズ (バイト数) が記されている。

 再生ストリームの任意の時刻へ飛び込み、再生を行う場合、タイムマップ情報の各エントリの時間情報を加算して対象となるVOBUを検出し、同様にVOBUのサイズ情報を加算した結果に基づいて、ファイル内でのシーク先を導くことが可能になる。

 このように、タイムマップ情報は、ストリーム内の時間情報とアドレス情報の変換を行うフィルタの役割を果たしている。

<ENTRY> エントリ情報

時間情報 (duration)

サイズ情報 (size)

図20は、字幕情報「subtitle1.xml」他、の詳細を記した図である。

タグ<SUBTITLE>に囲まれた下記の情報によって各言語毎の字幕が記述されている。

<LANGUAGE> 言語情報

言語情報 (language)

10 キャラクタ情報 (character)

フォント情報 (font)

カラー (color)

斜体 (italic)

太字 (bold)

15 下線 (underline)

言語情報タグの属性値として、英語、日本語などを示す言語情報、シフトJISなどのキャラクタ情報、明朝体などのフォント情報と、表示文字の修飾情報が記述される。

<TEXT> テキスト情報

20 表示開始時刻情報 (in)

表示終了時刻情報 (out)

テキスト

(ストリーム構造)

図21を用いてストリームの詳細を説明する。

25 本実施例で使用するストリームは、MPEG-2と呼ばれる国際規格ISO/IEC13818に基づいている。MPEG-2は、ビデオストリーム、オーディオストリームと、それらを多重化(一つのストリームに束ねる)するシステムストリームとから構成されている。

ビデオデータは、Iピクチャ(フレーム内符号化)、Pピクチャ(時間予測符

号化) およびBピクチャ(双方向時間予測符号化)を有するGOP構造に圧縮される。それぞれのピクチャ間での参照関係は図21の通りである。

圧縮されたビデオデータは、パケット化され、次いでパック化されてオーディオデータと多重化され、1本のシステムストリームを形成する。

- 5 多重化層では、GOPを基準(GOP先頭を含むパックから次のGOP先頭を含むパックの直前まで)としてVOBUを形成する。GOPはビデオ層で定義されている言葉であり、システム層での定義に当てはめることができないため、VOBUを導入している。

10 また、本実施例の中では、MPEG-2システムストリームのことをVOB(Video Object)という。
(プレーヤ構造)

図22は、映像再生装置のブロック構成図である。

- 映像再生装置は、STBなどの外部チューナからデータを受け取る受信手段
(101)、データを記録する蓄積媒体(102)、CPU(103)、プログラ
15 ムメモリ(104)、作業用ワークメモリ(105)、ストリームをデコード
するデコーダ(106)、モニタおよびスピーカへ表示出力する表示手段(10
7)、ユーザからの要求を受けるI/F(108)から構成されている。CPU
(103)はその内部にクロックを持ち、日時情報を有しており、また、作業用
ワークメモリ(105)には、映像再生装置の再生制御ステータス情報(fu
20 ll/restricted/free)が格納されている。

(クラスライブラリ詳細)

図23は、Eパッケージの映像再生装置のソフトウェア構成図である。

- オペレーティングシステム(203、以後OSと略する)を中心に、OSの下
には、ファイルシステムドライバ(201)と、デバイスドライバ(202)と
25 を有する。ファイルシステムドライバ(201)は、ディスク上のデータをファ
イルまたはディレクトリ構造を用いてアプリケーションへのアクセス環境を提供
する。デバイスドライバ(202)は、コンピュータのハードウェアデバイスで
あるデコーダやグラフィックカードを制御する。

また、OS上には、ミドルウェア(204)が乗る。例えばJavaの場合は、

J a v a の仮想マシン (V i r t u a l M a c h i n e、以下 J a v a V M と称する) とクラスライブラリが置かれる。このクラスライブラリとして、E パッケージ用のクラスライブラリ (205) も置かれる。

5 標準のクラスライブラリおよびEパッケージ用のクラスライブラリは、アプリケーションに対してクラスやそのメンバ関数などをプログラミング環境として提供する。

アプリケーションとして、Eパッケージ専用のプレーヤアプリケーション (206) の他にサードパーティから提供される外部アプリケーション (207) がそれぞれ動作する。

10 図24は、ミドルウェア内でのEパッケージクラス構成図である。

ミドルウェア内にはEパッケージ用として、「P a c k a g e クラス」「T i t l e クラス」「M e n u クラス」「A u d i o クラス」「S u b t i t l e クラス」「E v e n t クラス」「L i n k クラス」「C u r s o r クラス」「S t a t u s クラス」などが用意されている。以下、個々に説明する。

15 「P a c k a g e クラス」

P a c k a g e クラスは、最初に呼び出されるクラスである。パッケージ情報 p a c k a g e . x m l を基にインスタンスが生成される。

図25はP a c k a g e クラスの処理を示している。

20 コンストラクタ P a c k a g e (p a c k a g e) は、p a c k a g e . x m l を読み込み、P a c a k a g e インスタンスの属性値を取得する (2501)。前述したように、インスタンスの属性値は、全て対象の管理情報ファイル内に記述されている。

次に、リージョン情報 (r e g i o n)、レベル情報 (l e v e l) 日時情報 (e x p i r e) からパッケージの再生が可能か否かの認証処理を行う (2502)。再生不可 (p r o h i b i t e d) の場合は、アプリケーションに対してエラーが返され、終了する (2503)。

25 認証プロセスを通過 (p e r m i t t e d) した場合は、アップデートチェックを行う (2504)。

アップデートタグ (<U P D A T E>) の日時情報 (d a t e) をCPUの日

時情報と比較して、アップデート予告日を経過していた場合、且つ、自動アップデート情報 (a u t o) が“y e s”の場合、インターネットからのダウンロードを実行 (2505) して、新しい再生制御情報で再生を再開する (2501)。

5 アップデートチェック (2504) でダウンロードを実行しない場合、Menu
インスタンス生成 (2506)、Titleインスタンス生成 (2507) を
それぞれ実行する。

Packageインスタンスは、getMenu、getTitleメンバ
関数を有する。Packageインスタンス生成後、アプリケーションはこの関
数をコールして、Menuおよび各Titleインスタンスをそれぞれ入手する
10 ことができる。

「Titleクラス」

Titleクラスは、タイトルの再生制御を行うクラスである。各タイトル毎
にインスタンスが生成され、インスタンスをCanvasインスタンス上に配置
(a d d) することで画面への描画が行われる。また、メンバ関数のコールによ
15 ってタイトル再生が制御される。

図26および図33はTitleクラスの処理を示している。

コンストラクタTitle (t i t l e) は、起動されると同時にt i t l e .
xmlを読み込み (2601)、Link_LISTに基づきLinkリストを
内部に生成する (2602)。この時、対象タイトルがアクセス可能な場所にあ
20 るかをチェックし、アクセス不可能の場合はリストから削除する。具体的には、
ネットワークプロトコルなどを使って当該ファイルが存在するのを確認するが、
本発明とは直接関係しないので、詳細な説明は省略する。

次に、Chapterリストを生成 (2603) し、Chapterが参照す
るストリームの属性情報ファイル (例えば、stream.xml) を読み込み
25 (2604)、Audio、Subtitleインスタンスを生成する (260
5) 。

次に、TIMELINE情報に基づき、Timelineリストを生成 (26
06) し、INTERFACE情報に基づき、関数リストを生成 (2607) し、
最後にCursorインスタンスを生成 (2608) し、リモコン (I/F) か

らの要求に対応できるように準備する。

`Title`クラスは、色々なメンバ関数を持っている。

再生 (`play`)、停止 (`stop`)、再生速度設定 (`setRate`)、再生位置設定 (`setTime`) は、AV再生を直接コントロールする関数である。
 5 これらはデコーダが提供する機能をそのままアプリケーションに提供する。例えば、`play`は、アプリケーションからコールされると、関数の有効/無効をチェックし、有効の場合は、デコーダに再生開始を指示する。

例えば、アプリケーションから再生関数 (`play`) が呼ばれた場合を考える。
 この時、再生関数 (`play`) は、プレーヤの再生可能状態 (`full playback/restricted playback/free only`)
 10 と関数リストから当該関数の使用制限を比較する (2611)。当該関数を使用可能である場合は、関数の実行を開始する (2612)。一方、当該関数を使用不可能である場合は、関数の処理を中止する。

ここで、関数の有効/無効の関係を下記の表に整理する。

	level = full	restricted	free
Status = full playback	有効	有効	有効
restricted playback	無効	有効	有効
free only	無効	無効	有効

上記の表は、`Title`インスタンスの関数の有効無効だけでなく、パッケージ自身のレベル (`level`) に対して再生が可能かを判断する基準にも一致する。
 20

音声、字幕の制御は、タイトル内でのストリーム、即ち、言語毎の属性値を持つインスタンスを取得する `getAudio`、`getSubtitle` と、再生するストリームを設定する `setAudio`、`setSubtitle` とがある。

25 `getAudio` および `getSubtitle` は、`Title` コンストラクタで生成した `Audio` インスタンス、`Subtitle` インスタンスをそれぞれの返り値としてアプリケーションに渡す (2621)。アプリケーションは、ここで得られたインスタンスをそのまま `setAudio` または `setSubtitle` の引数として使い再生ストリームを設定する。

5 `setAudio`および`setSubtitle`関数では、最初に当該関数が有効かをチェックする(2631)。具体的には、映像再生装置の再生可能状態(`full playback/restricted playback/free only`)と関数リストの当該関数の使用制限を比較する。当該関数が使用可能である場合は、受け取ったインスタンスの属性値に従い、デコードに再生ストリーム属性値を設定する(2632)。一方、当該関数が使用不可能である場合は、関数の処理を中止する。

映像再生装置のステータスと関数との比較は前述した表と同じである。

10 `Title`クラスは、その他にもイベント処理起動用に`enableEvent`を、タイトル間リンク処理起動用に`enableLink`関数を持つ。

15 `enableEvent`関数は、`Title`情報で記されたタイムライン情報(<TIMELINE>)、即ち、分岐情報(<BRANCH>)、メッセージ情報(<Message>)及びイベントトリガ情報(<TRIGGER>)を処理する。`enableEvent`関数は、コールされると内部でスレッドを起動する(3301)。起動されたスレッドは、以下のループ処理を行い続ける。

20 再生時刻情報を監視し、タイムラインリストにある各イベント、例えば分岐情報(<BRANCH>)、メッセージ情報(<Message>)、または、イベントトリガ情報(<TRIGGER>)が指定する有効時刻になっているかをチェックする(3302)。有効時刻になっている場合は、映像再生装置の再生可能情報(`Status`)と比較(3303)して各イベントが当該映像再生装置で実行可能か否かを判断する。

各イベントが実行可能な場合は、イベントの種類がユーザからのリクエストを必要とする分岐(BRANCH)か否かを確認する(3304)。

25 分岐(BRANCH)の場合は、ユーザからのリクエスト受け付け(3305)へと進み、分岐(BRANCH)の有効時間(`out`によって指定)までリクエストを待ちつづけるようにループする(3306)。リクエストを受けずに有効時間(`out`)が切れた場合は、再びループの先頭(3302)まで戻る。有効時間(`out`)までにユーザからのリクエストを受け付けた場合は、分岐先タイトル`Title`インスタンス(`jump`で指定)を生成し、当該タイトルの

再生へと進む(3307)。

ステップ3304で、当該処理が分岐(BRANCH)で無い場合、即ち、メッセージ(MESSAGE)またはイベントトリガ(TRIGGER)と判断された場合は、ステップ3308へ進み、当該処理がメッセージ(MESSAGE)かイベントトリガ(TRIGGER)かを判断する。該処理がメッセージ(MESSAGE)の場合は、指定されたメッセージ情報(message)からTextインスタンスを生成し(3309)、TextインスタンスをCanvasインスタンス上に表示(add)する(3310)。また、メッセージ表示期間(out)まで待ち(3311)、表示期間終了(out)と同時にTextインスタンスを消去(Canvasインスタンスからdelete)し(3312)、ループ先頭(3302)まで戻る。

ステップ3308で、当該処理がTRIGGERと判断された場合は、Eventインスタンスを生成して(3313)、アプリケーションによってインプリメントされている関数(eventExec)を実行し(3314)、ループ先頭(3302)まで戻る。

「Menuクラス」

Menuクラスは、Titleクラスから派生したメニューを表示するクラスである。タイトルと同じMenu毎にインスタンス化され、Canvasクラスに置くことで、画面表示がされる。

図27は、Menuクラスの処理を示している。

コンストラクタMenu()は、メニュー情報ファイルmenu.xmlを読み込み(2701)、Titleクラスとしての処理(2702)を経て、メニューページを生成(2703)、先頭メニューページを表示(2704)し、Cursorからのイベントを処理するmenuThreadを起動する(2705)。

各ページとして表示する内容は、図17で説明したようにMENU情報内のメニューページ(<MENU_PAGE>)によって記述されている。メニューページ内のタイトル(<TITLE>)に基づいてボタンが作られ、画面上に表示される。

マルチページメニューの場合は、メンバ関数 `nextPage` によって次ページへ (2711)、`prevPage` によって前ページ (2721) の表示へ移動する。また、タイトルが選択された場合、`selectedTitle` によってアプリケーションに対して選択されたタイトル情報を通知する (2731)。

- 5 `menuThread` 関数では、スレッドを起動 (2741) し、`Cursor` インスタンスからのイベントを受け付ける (2742)。`Cursor` インスタンスからのイベントを受け付けると、イベントがタイトル選択であるかをチェックし (2743)、タイトル選択であった場合は、`selectedTitle` 関数を呼び出し (2744)、アプリケーションに対して、選択タイトルを通知する。
- 10

ステップ 2743 で、タイトル選択でなかった場合は、次にページ移動があったかをチェックし (2745)、真であったばあいは、ページ移動が次ページへか、前ページへかを判断し (2746)、それぞれ `nextPage` の起動 (2747)、`prevPage` の起動 (2748) を行う。

- 15 「`Audio` クラス」 および 「`Subtitle` クラス」

`Audio` クラスは、各オーディオストリーム毎に属性値を有するクラスである。タイトル内で、例えば使用可能なオーディオストリームが 2 本あれば、`Audio` クラスのインスタンスが 2 つ生成される。どちらから一方のインスタンスを `Title` クラスの `setAudio` 関数に設定することで、再生するオーディオストリームの設定が行われる。

20

図 28 は、`Audio` クラスの処理を示している。

コンストラクタ `Audio()` は、ストリーム属性情報ファイル `stream.xml` を読み込み (2801)、インスタンス内に属性値を確保する (2802)。

- 25 また、`Audio` クラスは、メンバ関数 `getLang` でインスタンスの持つ言語情報、即ちストリームの対応する言語と、`getCoding` でインスタンスの持つ圧縮情報、即ちストリームの圧縮方式 (2811) と、`getChs` でインスタンスの持つチャンネル情報、即ちストリームのチャンネル数 (2821) とをそれぞれアプリケーションに返す。

SubtitleクラスもTitleクラスと同様の機能を有する。

「Eventクラス」および「Linkクラス」

Eventクラスは、タイトル内でのイベント生成を行うクラスであり、Link
5 クラスは、タイトル内でのタイトル間リンク情報用イベントの生成を行うク
ラスである。

図29は、EventクラスおよびLinkクラスの処理を示している。

Eventクラスのコンストラクタは、引数を基にEvent属性を設定する
(2901)。

Eventクラスのメンバ関数execEventは、アプリケーションによ
10 ってオーバーライトされる関数である。即ち、execEventの起動はイベ
ントハンドラを起動する(2911)。execEventは引数にID(i
d)を持ち、このIDによってアプリケーションはどのイベントトリガ(TRI
GGER)がかけられたのかを識別し、以後の処理を分岐して行うことが可能で
ある。

15 Linkクラスのコンストラクタは、引数で与えられたTitleのインスタ
ンスを生成する。

Linkクラスのメンバ関数であるnotifyLinkは、execEve
ntと同じくアプリケーションにオーバーライトされる関数であり、この関数
を用いてアプリケーションへTitleインスタンスを渡し、イベント処理を実行
20 する。

「Cursorクラス」

Cursorクラスは、画面上でのカーソルを処理するクラスである。図34
を使って説明する。

Cursorクラスのコンストラクタは、最初に位置情報を生成/初期化し
25 (3401)、リモコンとの通信を開始し(3402)、カーソル処理のスレッ
ドCursorThreadを起動する(3403)。

カーソル処理スレッドCursorThreadでは、最初にスレッドを起こ
し(3411)、処理ループに入る。処理ループでは、カーソルの移動が起きた
かをチェックし(3412)、移動が起きた場合には、moved関数を呼び

(3413)、位置情報を更新する。移動がなかった場合、もしくはステップ3413の後に、選択実行が行われたか、言い換えれば、カーソルが選択しているボタンに対して、ユーザが選択実行を行ったかを検出し(3414)、選択実行が行われた場合は、selected関数を呼び(3415)、現在のTitleへ通知を行う。

moved関数では、指定された引数に基づき、位置情報を修正し(3421)、selected関数では、選択実行要求があったことをTitleインスタンスへ通知する(3431)。

「Statusクラス」

Statusクラスは、映像再生装置のステータスを表現するクラスである。このクラスは映像再生装置、またはシステムに対して一意に存在するものであって、その都度インスタンス化はされない。アプリケーションからはクラスとしてそのままアクセスが行われる。

アクセスは、アプリケーションがステータスを知るgetStatusや有効期間を知るgetPeriodなどが用意されている。

図35に示すように、Statusクラスはパラメータで指定される値に基づき、Status情報およびPeriod情報を内部に生成する(3501)。メンバ関数getStatusはアプリケーションにStatus情報を返し(3511)、メンバ関数getPeriodはPeriod情報をアプリケーションに返す(3521)。

「Frameクラス」および「Canvasクラス」

画面構成をするクラスである。Frameクラスは画面表示の元になるクラスで、例えばWindows OSで見られるウィンドウに相当する。Frameインスタンス内には動画再生を行うCanvasインスタンスが置かれる。

図36を用いてCanvasクラスを説明する。

Canvasクラスは、コンストラクタにおいて、映像データをオーバーレイ表示するFrameインスタンスを生成する(3601)。続いて、デコーダの初期化(3602)、オーバーレイ、即ちグラフィック機能の初期化(3603)を行う。デコーダの初期化処理および、グラフィック機能の初期化処理は、

下位のオペレーティングシステム（OS）や、ハードウェアに依存する処理であり、また、本発明とは本質的に関連がないので、詳細な説明は省略する。

Canvasインスタンスによって、画面上に描画するウィンドウが表示されるが、実際の画像描画は、メンバ関数addによって実行される。関数addは
5 Titleインスタンスを引数として呼ばれ、Titleインスタンスが持つ、
ストリーム情報の読み出し（3611）、デコーダの設定（3612）を行う。
続いてデコーダにデコード処理を開始（3613）させ、オーバーレイにデコード
画像の描画を開始（3614）させる。

また、Canvasクラスは、メンバ関数setSizeを持ち、この関数に
10 よってCanvasのサイズを変更できる。内部の処理としては、Frameイ
ンスタンスのサイズ変更（3621）と、オーバーレイする表示サイズの変更
（3622）をそれぞれ行っている。

（プレーヤ再生処理）

次に、プレーヤアプリケーションとしての再生処理について説明する。

15 図30はプレーヤ再生処理のフローである。

プレーヤアプリケーションは、起動後（3001）、下記のようにCanvas
インスタンスを生成して、ビデオ表示ウィンドウを生成する（3002）。C
anvasインスタンス生成の内部動作は図36で説明した通りである。

```
Canvas objCanvas = new Canvas () ;
```

20 上記の記述は、Java言語に基づいている。左端のCanvasはクラス宣
言であり、続くobjCanvasがCanvasクラスのオブジェクト（イン
スタンス）であることを宣言している。また、new Canvas () は、C
anvasクラスのコンストラクタを呼び、これによって、objCanvas
が生成される。

25 次にユーザからのパッケージ選択を待ち（3003）、パッケージ選択後に、
下記のようにPackageインスタンスの生成（3004）、メニューインス
タンスの取得（3005）を行い、メニューを表示する（3006）。また、パ
ッケージインスタンスの生成は、図25で説明した通りである。

```
Package objPackage = new Package
```

```
(package) ;
```

```
Menu objMenu = objPackage. getMenu () ;
```

```
objCanvas. add (objMenu) ;
```

5 メニューは図31に示すように、背景画像とタイトル情報表示(テキスト)から構成されている。タイトルの選択(3007)は、カーソルをリモコンで移動させ、目的のタイトル上で「選択」を行うことによって実施される。

カーソルは、リモコンキー(上下左右)によって移動され、カーソルの移動や、選択実行は、図34で説明したように、スレッドとして起動されているCurs
10 orThread関数で検出および処理される。

例えば、次ページの移動が選択された場合は、Cursorインスタンスのselected関数が呼ばれ、Menuインスタンスはページ移動要求があったことを知る。次にMenuインスタンスはnextPage関数をコールして、メニューを次ページへと送る。

15 また、カーソルがタイトル4上で「選択」された場合、Cursorインスタンスの(selected)を通してタイトルが選択されたことをMenuインスタンスは取得する。次いで、selectedTitle関数がアプリケーションに対してタイトルが選択をされたことを通知し、アプリケーションはタイトル再生のステップ(3008以降)へと移行する。

20 プレーヤアプリケーションは、選択されたタイトル情報を引数に、PackageインスタンスのgetTitle関数をコールし、Titleインスタンスを取得する(3008)。次いで、プレーヤアプリケーションは、取得したTitleインスタンスのplay関数をコールしてタイトル再生を開始し(3009)、enableEvent関数をコールしてイベント用スレッドを起動する
25 (3010)。

```
Title objTitle = objPackage. getTitle (title) ;
```

```
objTitle. play () ;
```

```
objTitle. enableEvent () ;
```

以後、タイトル再生終了まで、イベント発生の確認（3011）、イベント発生時のイベント処理（3012）、タイトル再生終了の確認（3013）を繰り返す。タイトル再生終了が確認されたら、プレーヤアプリケーションでの処理を終了する（3014）。

5 また、ステップ3012のイベント処理は、図33で説明した通りである。

図32を用いてタイトル再生中のタイトル間ジャンプ処理を説明する。

タイトル1の中には図示するようにタイトル2への分岐受付期間が設けられている。分岐受付期間は、タイトル1の情報ファイル`title1.xml`内の`TIMELINE`情報内の`BRANCH`タグとその属性値で定義されている。

10 この分岐受付期間中、図32下に示すようにメッセージが表示され、ユーザが「選択」キーを押すとリンク先のタイトル2へと遷移する。

分岐受付期間（分岐情報（`<BRANCH>`）の属性値`in`から`out`で指定される区間）に入ると（図33、3302）、当該分岐（`BRANCH`）処理の`Status`と映像再生装置の持つ`Status`（`Status.getStatus()`によって取得）を比較して（前述した表の通り）、処理可能かを判断し（図33、3303）、`BRANCH`であることの確認（図33、3304）を経てユーザからのリクエスト待ちループ（図33、3305および3306）に入る。

ユーザからの選択実行リクエストは、`Cursor`インスタンスを通して受け取る（図34、3414から3415）。ユーザからの選択実行リクエストがあった場合は、新たな`Title`インスタンスを生成して、次のタイトル（図32ではタイトル2）の再生を開始する（図34、3307）。また、分岐受付期間（`out`）までにユーザからの選択実行リクエストがなかった場合には、タイムアウトの検出を経て（図33、3306）、分岐（`BRANCH`）処理を終了する。

25 図37は、ゲームアプリケーションの例である。本例では、プレーヤアプリケーションの代わりにゲームアプリケーションを起動する（3701）。プレーヤアプリケーションと同様に、`Canvas`インスタンスを生成して、ビデオ表示ウィンドウを生成する（3702）。`Canvas`インスタンス生成の内部動作

は図36で説明した通りである。

```
Canvas objCanvas = new Canvas () ;  
ゲームアプリケーションはゲームを開始し (3703)、ゲームアプリケーション  
5  が使用するPackageインスタンスの取得 (3704) とTitleイ  
ンスタンスの取得 (3705) を行う。取得したTitleインスタンスのpl  
ay関数をコールしてタイトル再生を開始し (3706)、enableEve  
nt関数をコールしてイベント用スレッドを起動する (3707)。
```

```
Package objPackage = new Package  
(package) ;  
10 Title objTitle = objPackage. getT  
itle (title) ;  
objTitle. play () ;  
objTitle. enableEvent () ;
```

以後、ゲーム終了まで、イベント発生の確認 (3708)、イベント発生時の
15 イベント処理 (3709)、タイトル再生終了の確認 (3710) を繰り返す。
ゲーム終了が確認されたら、ゲームアプリケーションの処理を終了する (371
1)。

ゲームアプリケーションの場合、イベントトリガを使ってゲームとAV再生を
同期させることができる。例えば、図18で説明したタイトル情報 (<TITLE
20 E>) 内のタイムライン情報 (<TIMELINE>) に、以下のようなイベン
トリガ (<TRIGGER>) を入れておく。

```
<TRIGGER level="full" id="1" event  
="1"
```

```
time="00:01:00:00"/>
```

25 時刻00:01:00:00 (1分) になると、イベントスレッドは対象時刻
(time) になったことを認識し (図33、3302)、ステータス (Sta
tus) 確認 (図33、3303)、分岐 (Branch) 確認 (図33、33
04)、メッセージ (Message) 確認 (図33、3308) を経て、Ev
ent生成 (図33、3313)、execEvent起動 (図33、331

4) が実行される。

起動されるメンバ関数 `execEvent` は、ゲームアプリケーションによってオーバーライトされており、ゲームアプリケーションは、`execEvent` から得た `id` に基づいて、ゲーム側での同期処理を行うことが可能である。

5 図38および図39は、パッケージと映像再生装置との間でのステータス (`Status`) または有効期限 (`Expire`) による再生制限を解除するためにネットワークを介してサーバから `Status` または `Expire` 情報を更新する例を説明した図である。

10 図25で説明した通り、パッケージの持つ再生レベル (`level`) および有効期限 (`expire`) と、映像再生装置が持つステータス (`Status`) および日時情報をそれぞれ比較して、パッケージを再生できるかを認証 (図25、2502) する。再生可能であれば2504以降のステップを処理し、再生不可能であればステップ2503によって再生処理を終了させている。

15 図25の再生処理終了 (2503) に代わって、映像再生装置のステータス (`Status`) またはパッケージの有効期限 (`expire`) を更新する処理を行ってもよい。

図39は、映像再生装置のステータス (`Status`) がパッケージのレベル (`level`) に合わなかったため、再生が出来なかった場合に、サーバと通信を行い、映像再生装置のステータス (`Status`) を更新する例である。

20 図25の終了 (2503) に代わって、図39では、ステータス (`Status`) 更新 (2503) が実行される。最初にアップデート・アプリケーションを起動する (250301)。このアップデート・アプリケーションは、プレーヤアプリケーションまたはゲームアプリケーション同様にミドルウェア上に築かれた一つのアプリケーションでもよく、あるいはオペレーティングシステム上で直接起動されるバイナリコードベースのアプリケーションであってもよい。例えば、

25 ミドルウェア上のアプリケーションであった場合は、プレーヤアプリケーションが `Loader` クラスを介してアップデート・アプリケーションを起動することができる。

アップデート・アプリケーションは、ミドルウェア (`Java`) で提供される

Socketクラスや直接ネットワークプロトコル(TCP/IPなど)を使って、サーバと通信をする(250302)。通信するサーバは、Packageの<INTERNET URL=""/>によって指示されている。アプリケーションは、サーバと通信して、ステータス(Status)更新をするために必要な条件(金額)を入手し(250303)、ユーザに提示する(250304)。

アプリケーションは、ユーザからのレスポンスを待ち(250305)、ユーザがステータス(Status)の更新を望む場合(250306)は、サーバとの課金処理(250307)を行い、ステータス(Status)更新処理(250308)を行い、アプリケーションを終了し(250309)、プレーヤアプリケーションは図25のステップ2501からの処理を再び行う。

上記課金処理は、クレジットカード番号の入力と通信などによって行う。インターネット上での課金システムには様々な技術があり、また、本発明とは本質的に関係がないので、詳細な説明は省略する。

ステップ250306で、ユーザがステータス(Status)の更新を望まない場合は、そのまま終了する(250310)。

ここでは、ステータス(Status)の更新を例に説明をしたが、有効期限(expire)の更新も同様にして行うことができる。ただし、この場合は、映像再生装置のステータス(Status)情報を更新するのではなく、パッケージの期限(expire)情報を更新する。パッケージがReWritableメディアによって記録されている場合は、直接情報を更新することが可能である。一方読み取り専用メディアの場合は、映像再生装置が備えた不揮発性メモリまたはハードディスクなどの一時記録媒体に期限(expire)情報を一時的に記録するシステムを設けることで、更新情報を再利用することが可能である。

本発明の映像再生システムは、映画を単純に再生する映像再生装置としてだけでなく様々なアプリケーションを実現することを目的としている。このため、この映像再生装置では、内部メモリに予め読み込まれ実行されるソフトウェアとして、オペレーティングシステムの種類に対応した機能の差異を吸収するミドルウェアを含んでいる。このミドルウェアは、プレーヤアプリケーションが映像コンテンツを再生したり、ゲームアプリケーション等の拡張アプリケーションを実行

5 するために用いられるツールを含むクラスライブラリを有している。具体的には、このミドルウェアは、上述のようにEパッケージ用のクラスライブラリを有している。ここでツールとは、各機能を実現するために用いられるクラス及びそのメソッド関数等である。また、パッケージメディアに含まれる再生制御情報（管理情報）に記録されている機能リストには、プレーヤアプリケーションやゲームアプリケーションを含むアプリケーションにクラスライブラリによって提供される機能を記述している。また、この機能リストは各機能毎のステータス情報を有し、映像再生システム自身が有するステータス情報との比較によって、各映像再生システムで再生可能なコンテンツを機能レベルでコントロールすることが可能になる。

10 この結果、Eパッケージで実現する様々なアプリケーションを、ビジネスやサービスの形態や質に応じて制御することが可能になる。

15 上述の通り、本発明は好ましい実施の形態により詳細に説明されているが、本発明はこれらに限定されるものではなく、特許請求の範囲に記載された本発明の技術的範囲内において多くの好ましい変形例及び修正例が可能であることは当業者にとって自明なことであろう。

請 求 の 範 囲

1. 外部から供給されるパッケージメディアを再生する映像再生装置であって、
前記パッケージメディアは、

5 映像データと、前記映像データの再生を制御する再生制御情報とを所定のデータフォーマットで格納する映像コンテンツと、

前記映像コンテンツを使用する拡張アプリケーションソフトウェアとを含み、
前記映像再生装置は、

内部メモリに予め保持され実行されるソフトウェアとして、

10 複数の種類のオペレーティングシステムから選ばれた一つのオペレーティングシステムと、

前記オペレーティングシステムの種類に対応した機能の差異を吸収するミドルウェアと、

15 前記ミドルウェア上で動作し、前記映像コンテンツを再生するプレーヤアプリケーションソフトウェアとを備え、

前記ミドルウェアは、

前記プレーヤアプリケーションソフトウェアが前記パッケージメディアを再生したり前記拡張アプリケーションソフトウェアを実行するために用いるツールを含むクラスライブラリを有し、

20 前記プレーヤアプリケーションソフトウェアは、前記ミドルウェアの前記クラスライブラリに含まれる前記ツールを介して、前記パッケージメディアの前記映像コンテンツを、前記所定のフォーマットに従って統一的に再生し、

25 前記拡張アプリケーションソフトウェアは、同一のパッケージメディアに含まれる前記映像コンテンツを用いて、前記ミドルウェアの前記クラスライブラリに含まれる前記ツールを介して実行される、映像再生装置。

2. 前記映像再生装置は、再生ステータス情報を管理しており、前記パッケージメディアの前記再生制御情報は前記再生ステータス情報に対応する再生制限情報を含み、前記再生制御情報を解析し、前記再生制御情報内の再生制限情報と前記再生ステータス情報の比較により、前記拡張アプリケーションソフトウェアが

前記ミドルウェアの前記クラスライブラリに含まれる前記ツールのいずれかを無効に設定する、請求項 1 記載の映像再生装置。

3. 外部から供給されるパッケージメディアを映像再生装置で再生する映像再生方法であって、

5 前記パッケージメディアは、

映像データと、前記映像データの再生を制御する再生制御情報とを所定のデータフォーマットで格納する映像コンテンツと、

前記映像コンテンツを使用する拡張アプリケーションソフトウェアとを含み、前記映像再生方法は、

10 複数の種類のオペレーティングシステムから選ばれた一つのオペレーティングシステムを前記映像再生装置の内部メモリに読み込み、起動するステップと、

該オペレーティングシステムの種類に対応した機能の差異を吸収するミドルウェアであって、該ミドルウェア上で動作するアプリケーションソフトウェアが前記パッケージメディアを再生したり、実行するために用いるツールを含むクラスライブラリを有するミドルウェアを前記映像再生装置の前記内部メモリに読み込み、起動するステップと、

15 前記ミドルウェア上で動作し、前記映像コンテンツを再生するプレーヤアプリケーションソフトウェアを前記映像再生装置の内部メモリに読み込み、起動するステップと、

20 前記ミドルウェア上で動作し、前記映像コンテンツを使用する拡張アプリケーションソフトウェアを前記映像再生装置の内部メモリに読み込み、起動するステップと、

前記プレーヤアプリケーションソフトウェアによって、前記ミドルウェアの前記クラスライブラリに含まれるツールを介して、前記パッケージメディアの前記映像コンテンツを、前記所定のフォーマットに従って統一的に再生するステップと、

25 前記拡張アプリケーションソフトウェアによって、前記映像コンテンツを用いて、前記ミドルウェアの前記クラスライブラリに含まれる前記ツールを介して実行するステップと

を含む映像再生方法。

4. 外部から供給されるパッケージメディアを再生する映像再生プログラムであって、

前記パッケージメディアは、

5 映像データと、前記映像データの再生を制御する再生制御情報とを所定のデータフォーマットで格納する映像コンテンツと、

前記映像コンテンツを使用する拡張アプリケーションソフトウェアとを含み、前記映像再生プログラムは、

内部メモリに予め保持され実行されるソフトウェアとして、

10 複数の種類のオペレーティングシステムから選ばれた一つのオペレーティングシステムと、

前記オペレーティングシステムの種類に対応した機能の差異を吸収するミドルウェアと、

15 前記ミドルウェア上で動作し、前記映像コンテンツを再生するプレーヤアプリケーションソフトウェアとを備え、

前記ミドルウェアは、

前記プレーヤアプリケーションソフトウェアが前記パッケージメディアを再生したり前記拡張アプリケーションソフトウェアを実行するために用いるツールを含むクラスライブラリを有し、

20 前記プレーヤアプリケーションソフトウェアは、前記ミドルウェアの前記クラスライブラリに含まれる前記ツールを介して、前記パッケージメディアの前記映像コンテンツを、前記所定のフォーマットに従って統一的に再生し、

25 前記拡張アプリケーションソフトウェアは、同一のパッケージメディアに含まれる前記映像コンテンツを用いて、前記ミドルウェアの前記クラスライブラリに含まれる前記ツールを介して実行される、映像再生プログラム。

5. 請求項4に記載の前記映像再生プログラムを格納するコンピュータ読み取り可能な記録媒体。

6. 映像再生装置に外部から供給され、該映像再生装置で再生されるパッケージメディアであって、

前記パッケージメディアは、

映像データと、前記映像データの再生を制御する再生制御情報とを所定のデータフォーマットで格納する映像コンテンツと、

5 前記映像コンテンツを使用する拡張アプリケーションソフトウェアとを含み、
前記映像再生装置は、

内部メモリに予め保持され実行されるソフトウェアとして、

複数の種類のオペレーティングシステムから選ばれた一つのオペレーティングシステムと、

10 前記オペレーティングシステムの種類に対応した機能の差異を吸収する
ミドルウェアと、

前記ミドルウェア上で動作し、前記映像コンテンツを再生するプレーヤアプリケーションソフトウェアとを備え、

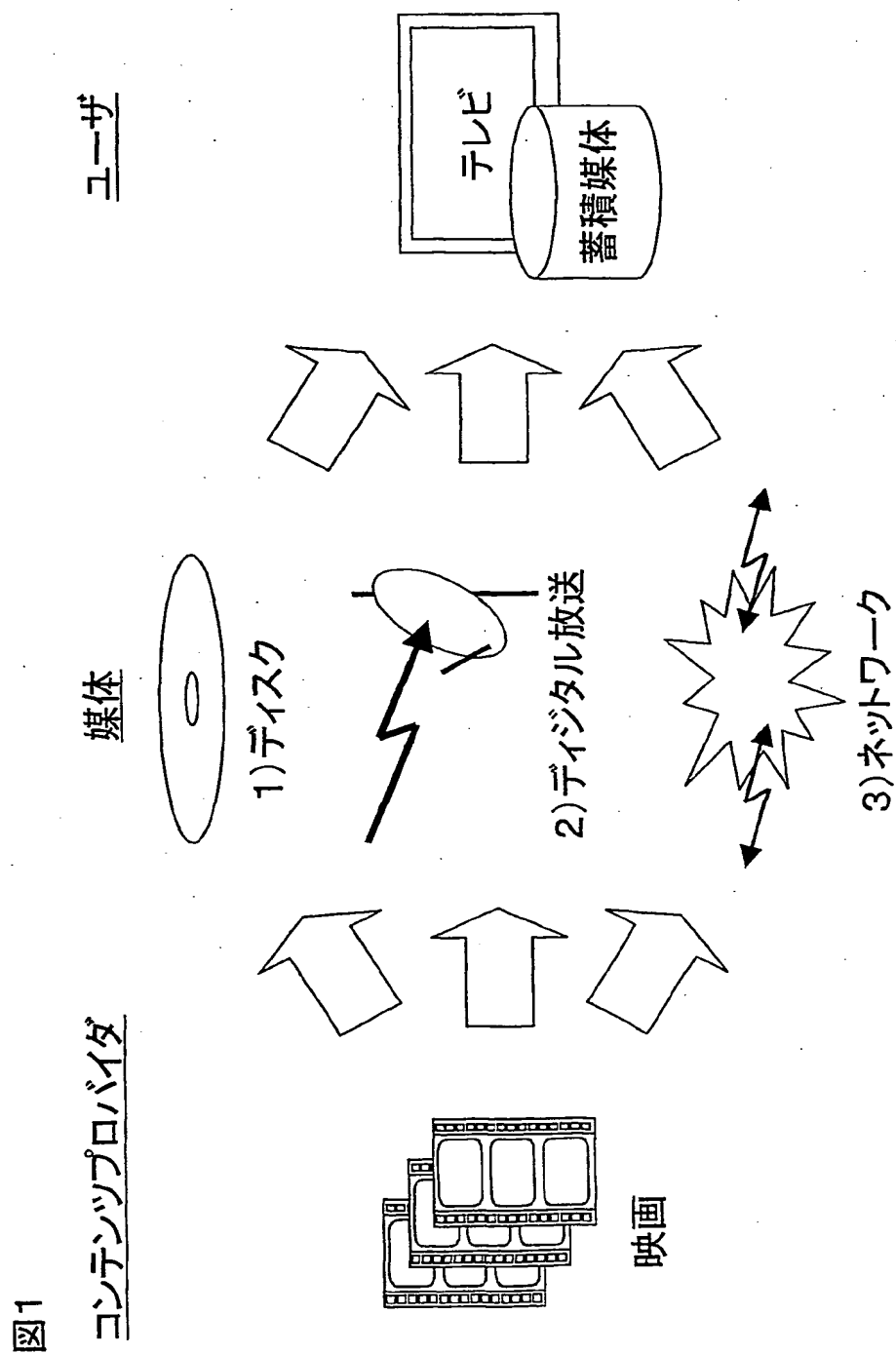
前記ミドルウェアは、

15 前記プレーヤアプリケーションソフトウェアが前記パッケージメディアを再生したり前記拡張アプリケーションソフトウェアを実行するために用いるツールを含むクラスライブラリを有し、

前記プレーヤアプリケーションソフトウェアは、前記ミドルウェアの前記クラスライブラリに含まれる前記ツールを介して、前記パッケージメディアの前記映像コンテンツを、前記所定のフォーマットに従って統一的に再生し、

20 前記拡張アプリケーションソフトウェアは、同一のパッケージメディアに含まれる前記映像コンテンツを用いて、前記ミドルウェアの前記クラスライブラリに含まれる前記ツールを介して実行される、パッケージメディア。

1 / 39



2 / 39

図2

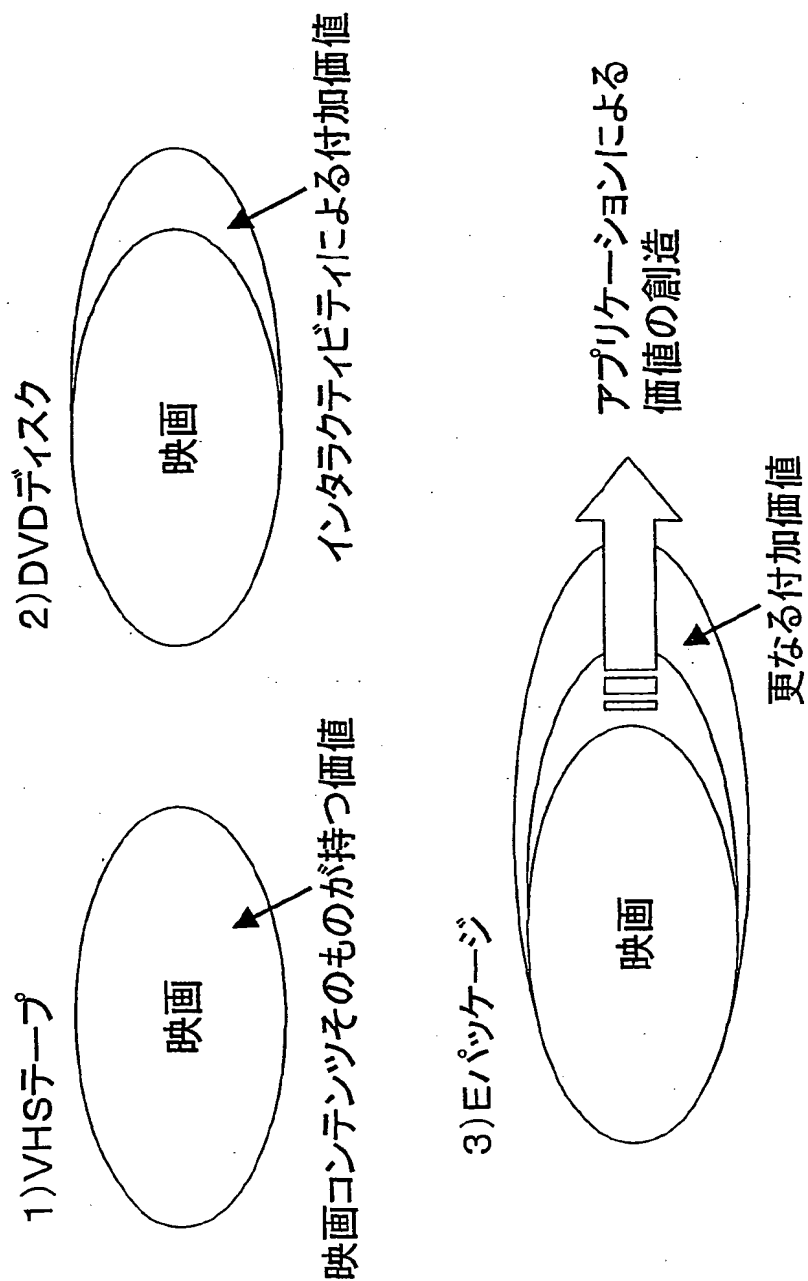
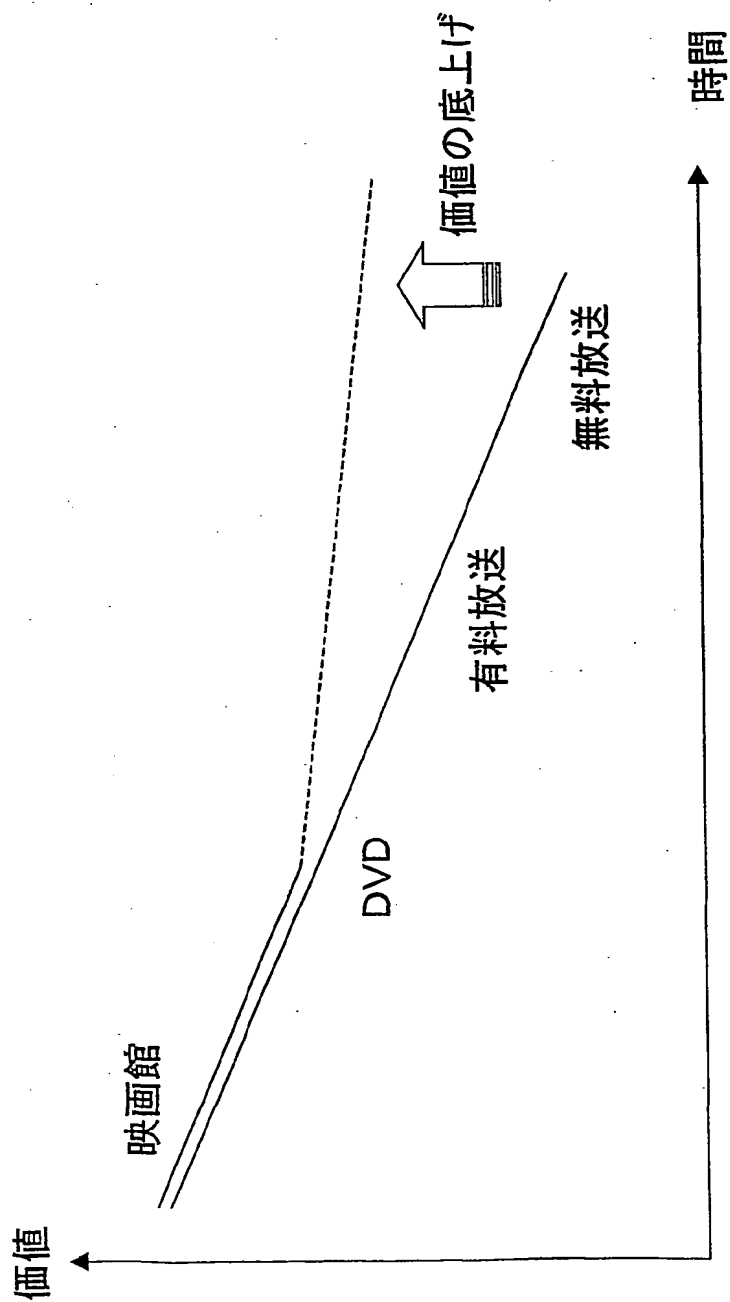


図3



4 / 39

図4

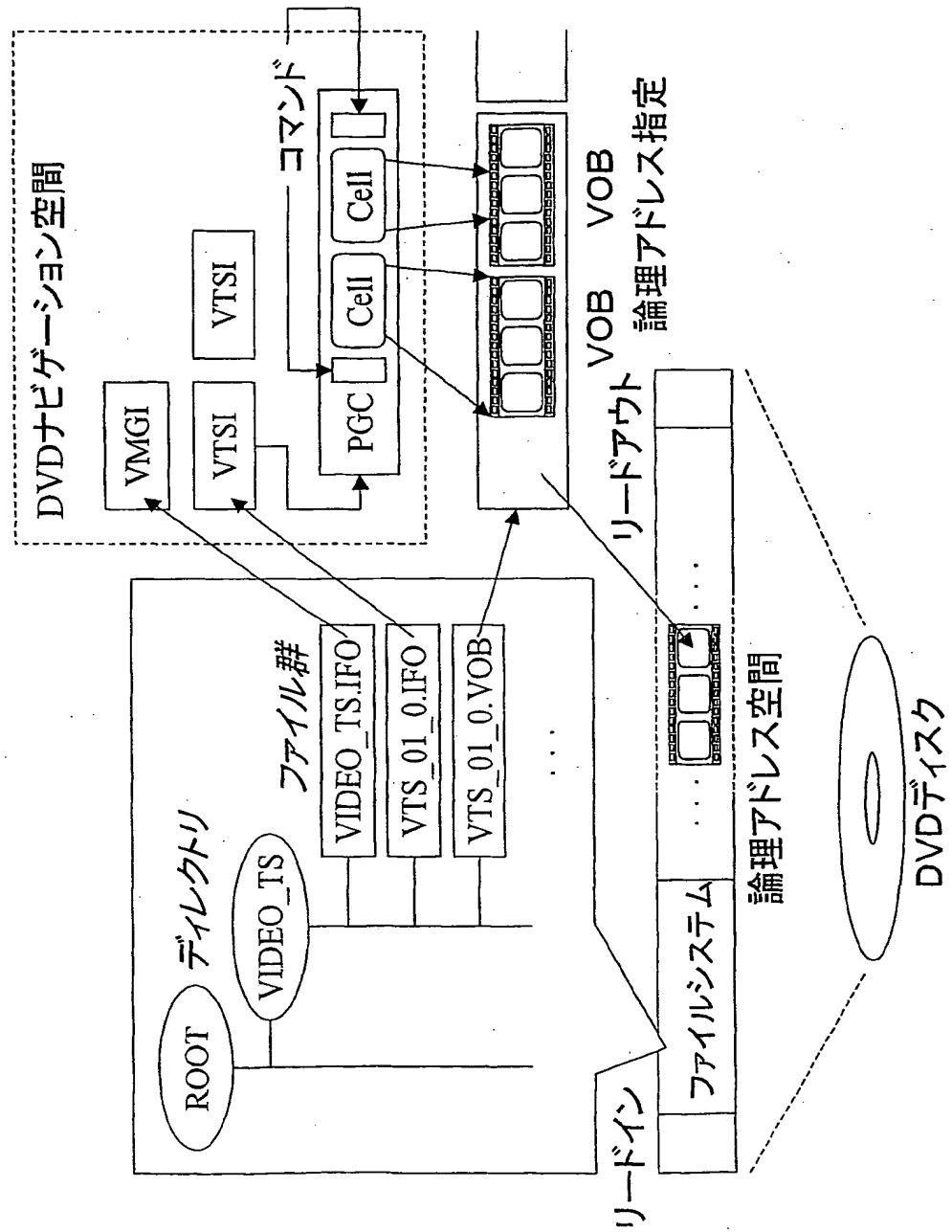
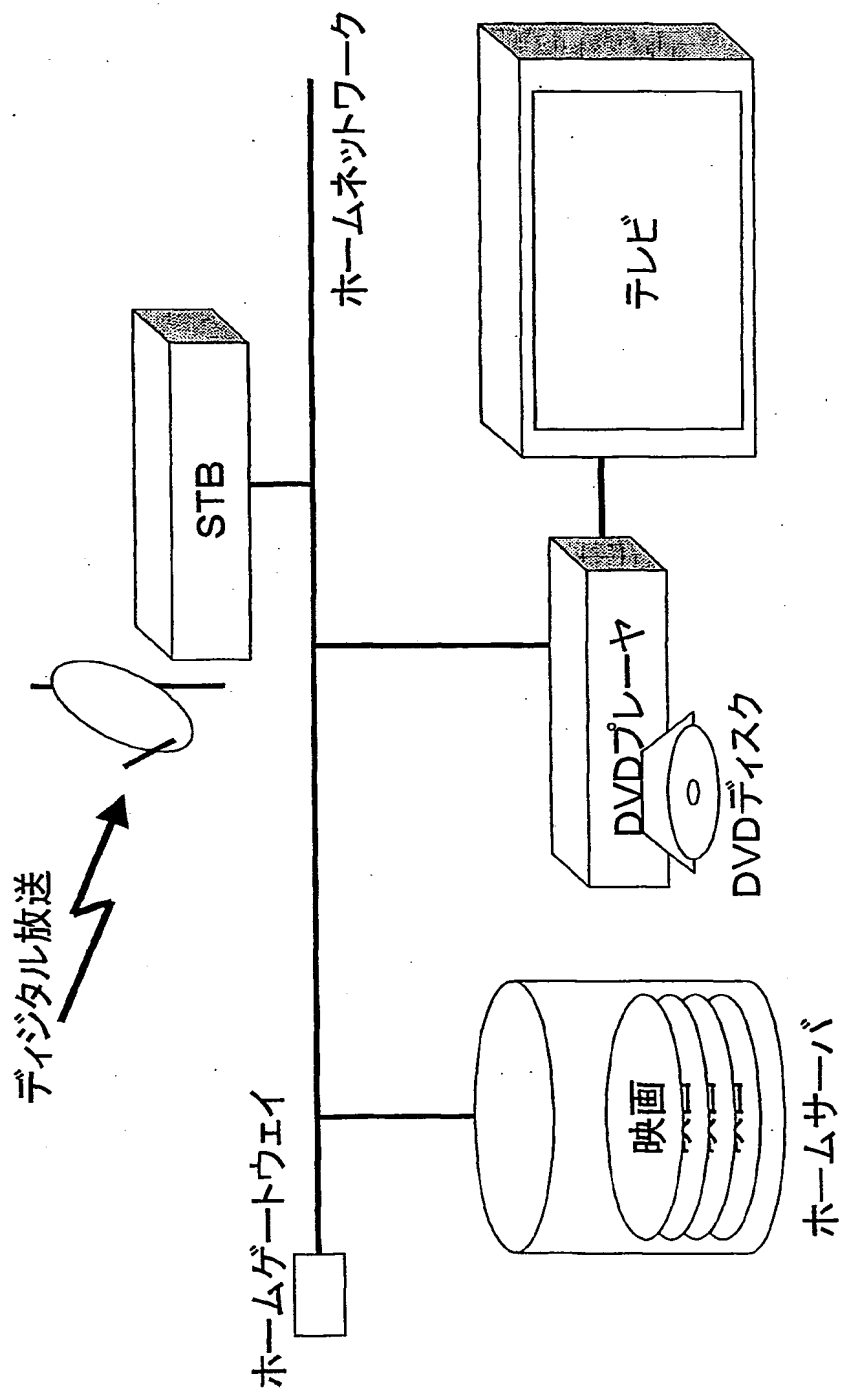


図5



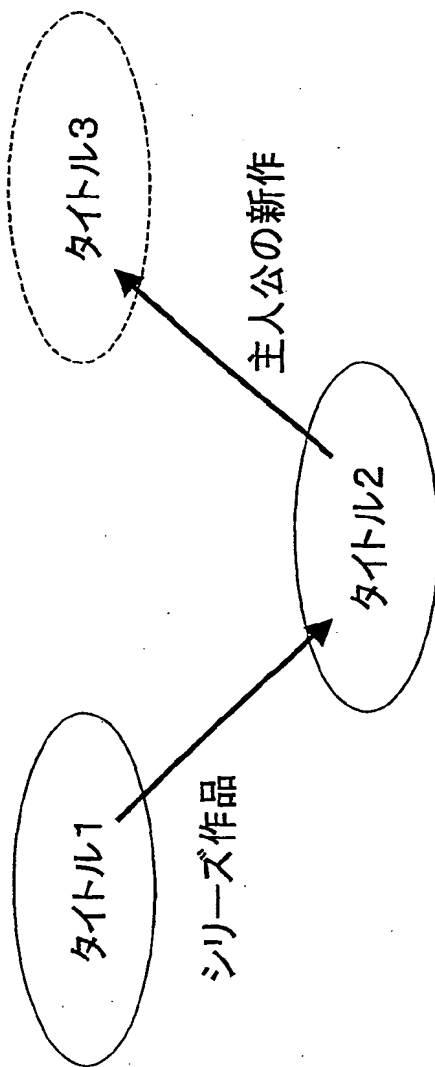


図6

7 / 39

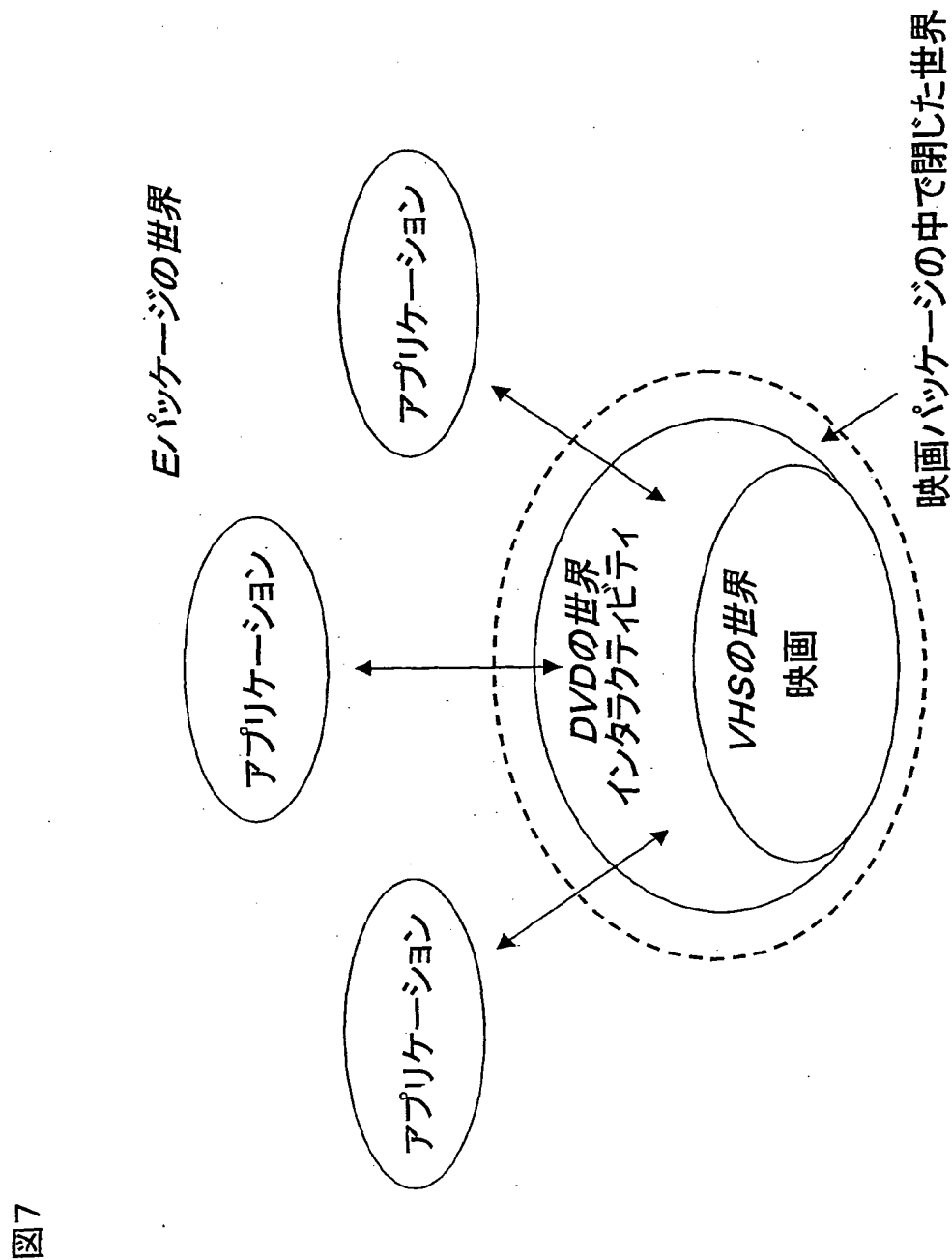


図8

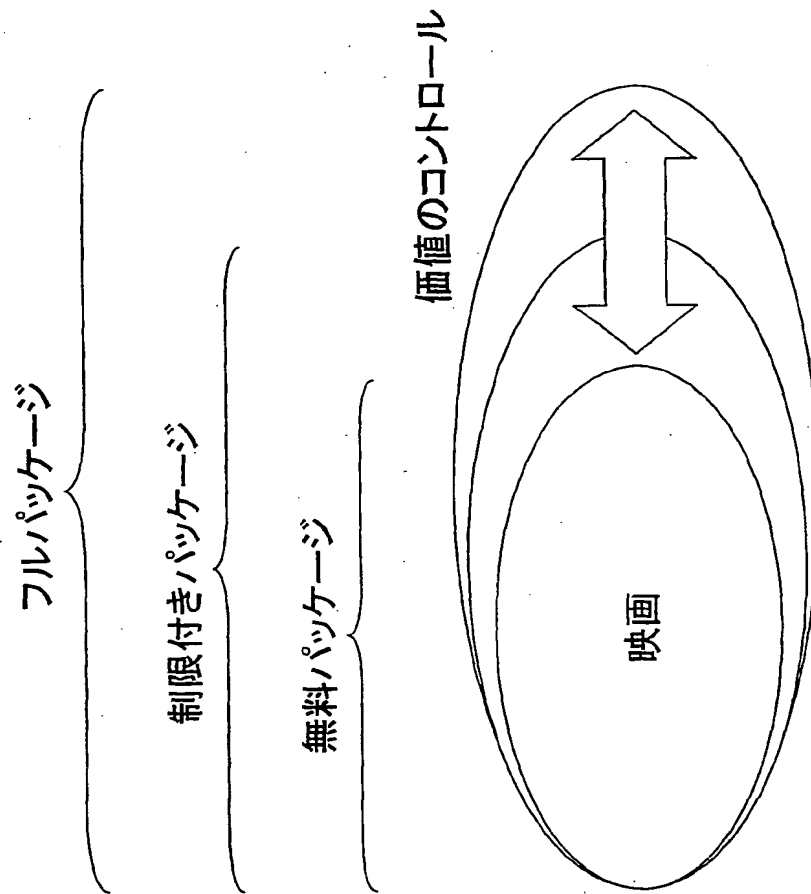
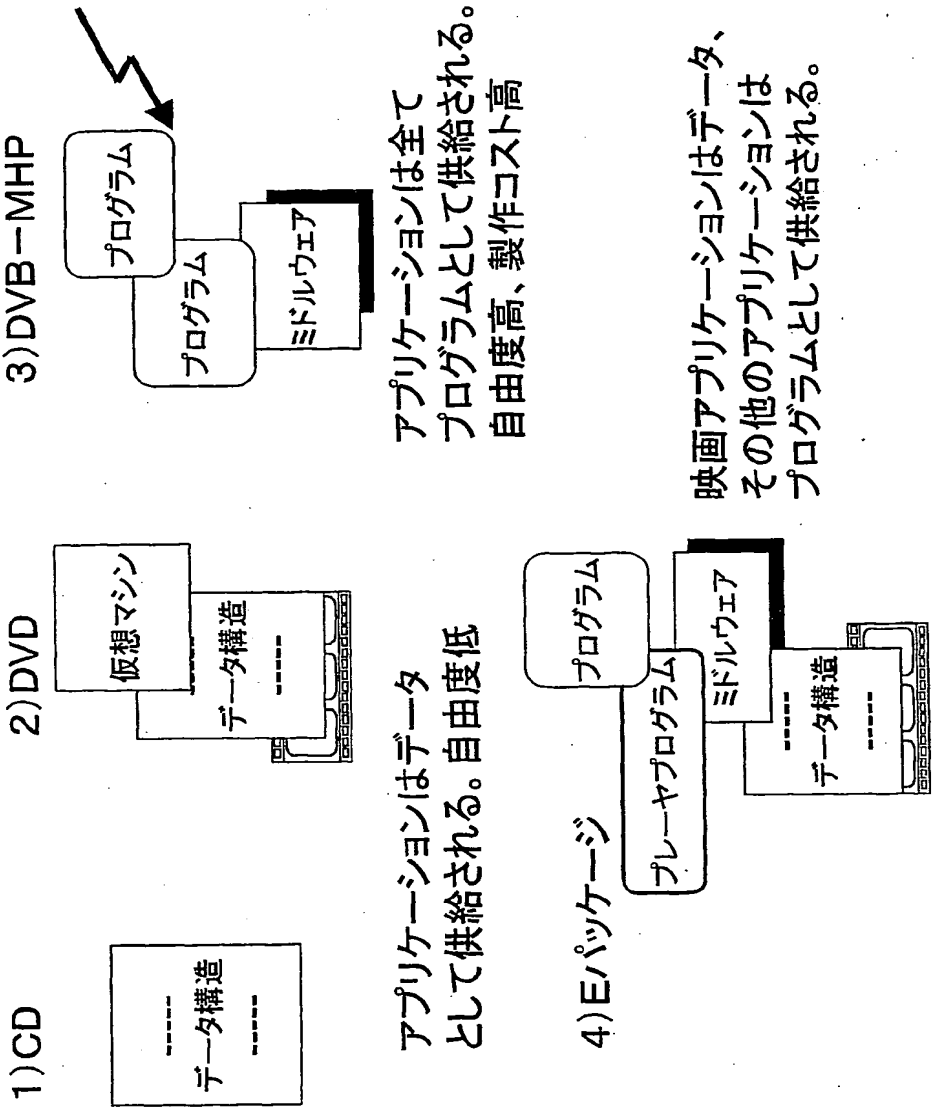
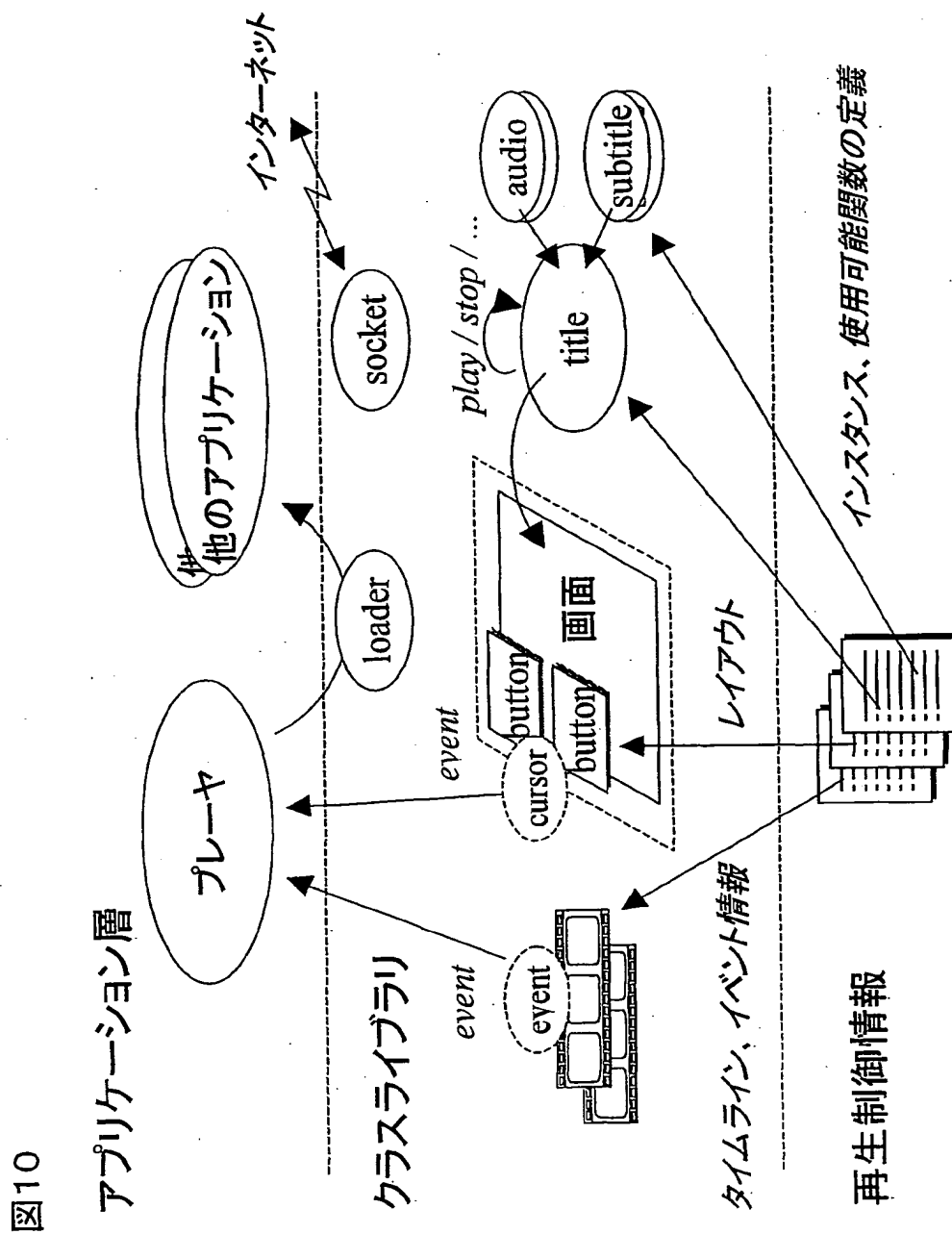


図9





11 / 39

図11

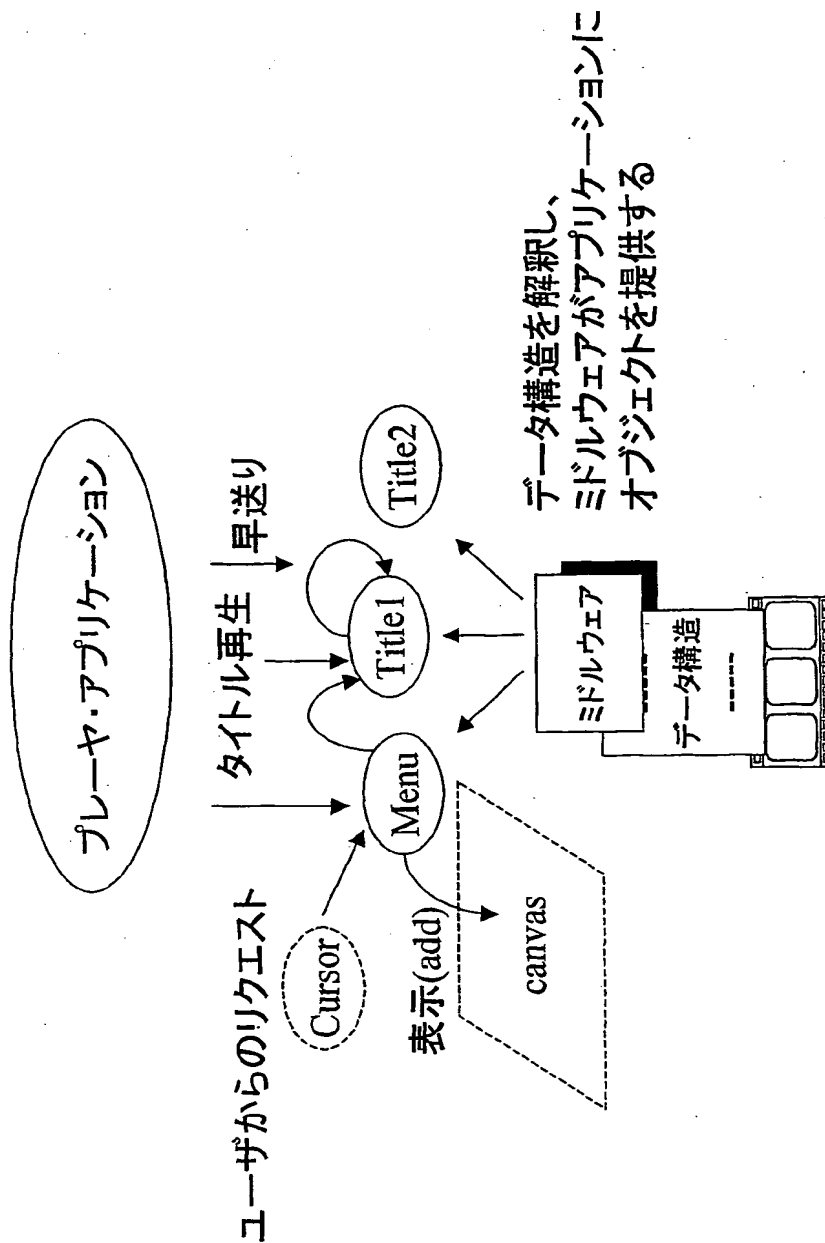
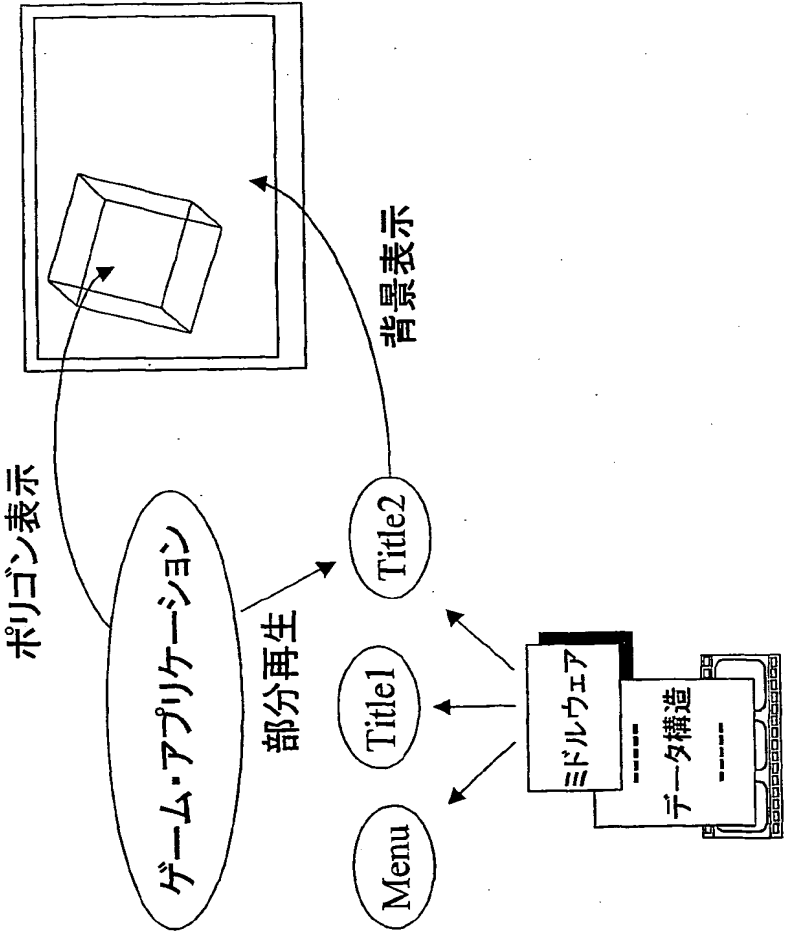


図12



13 / 39

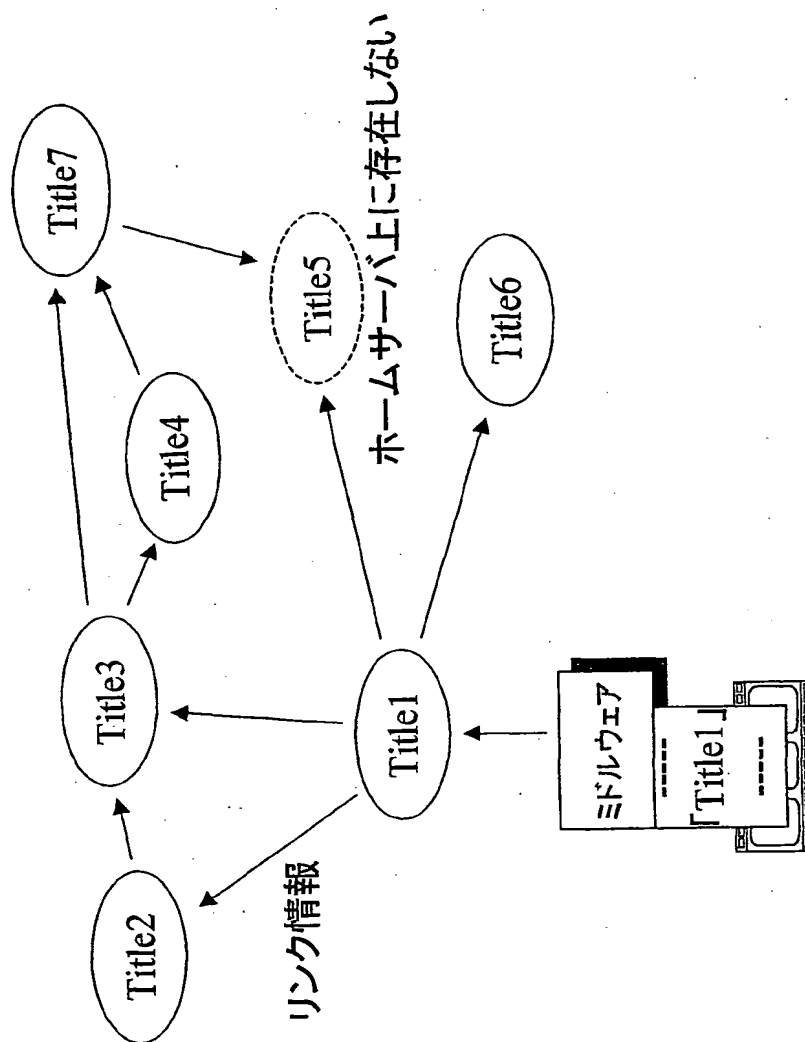
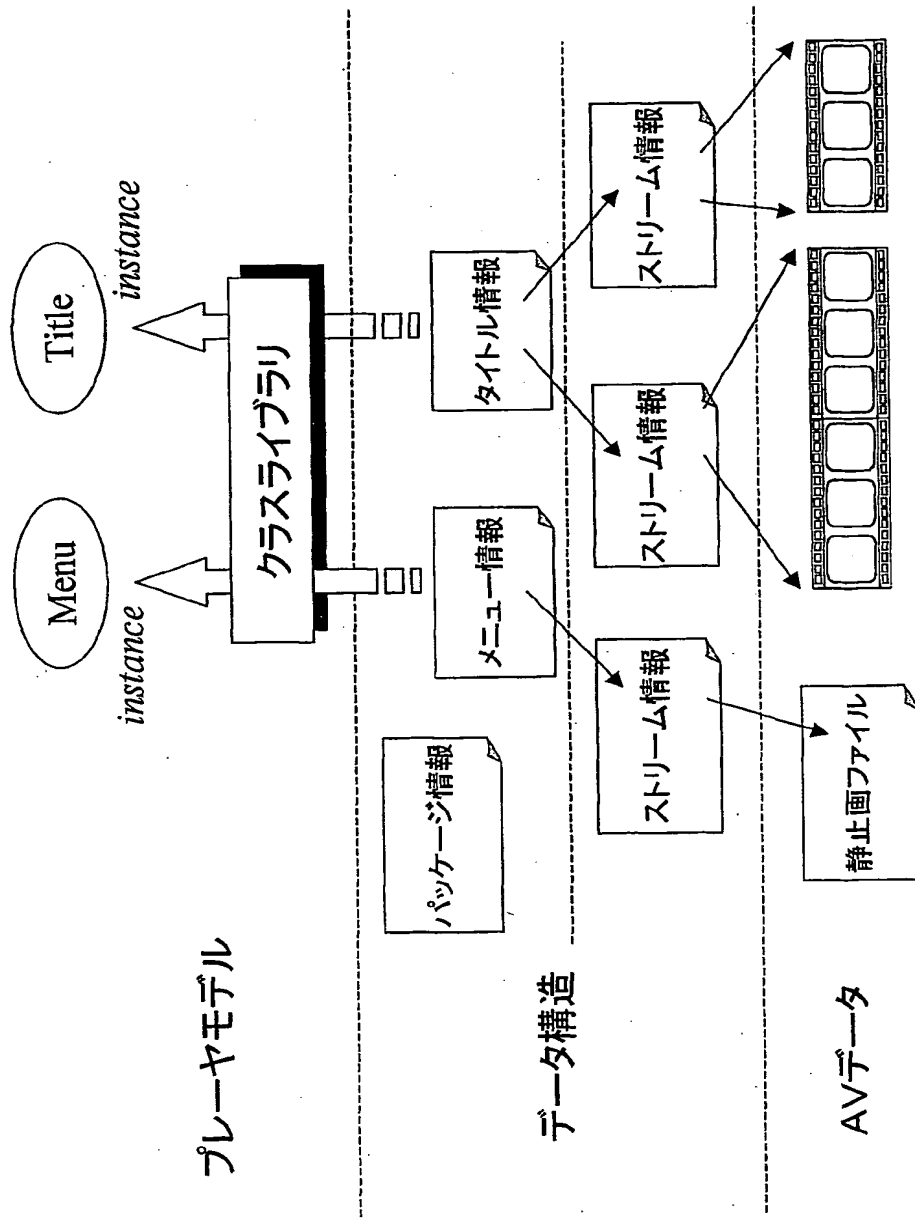


図13

14 / 39

図14



15 / 39

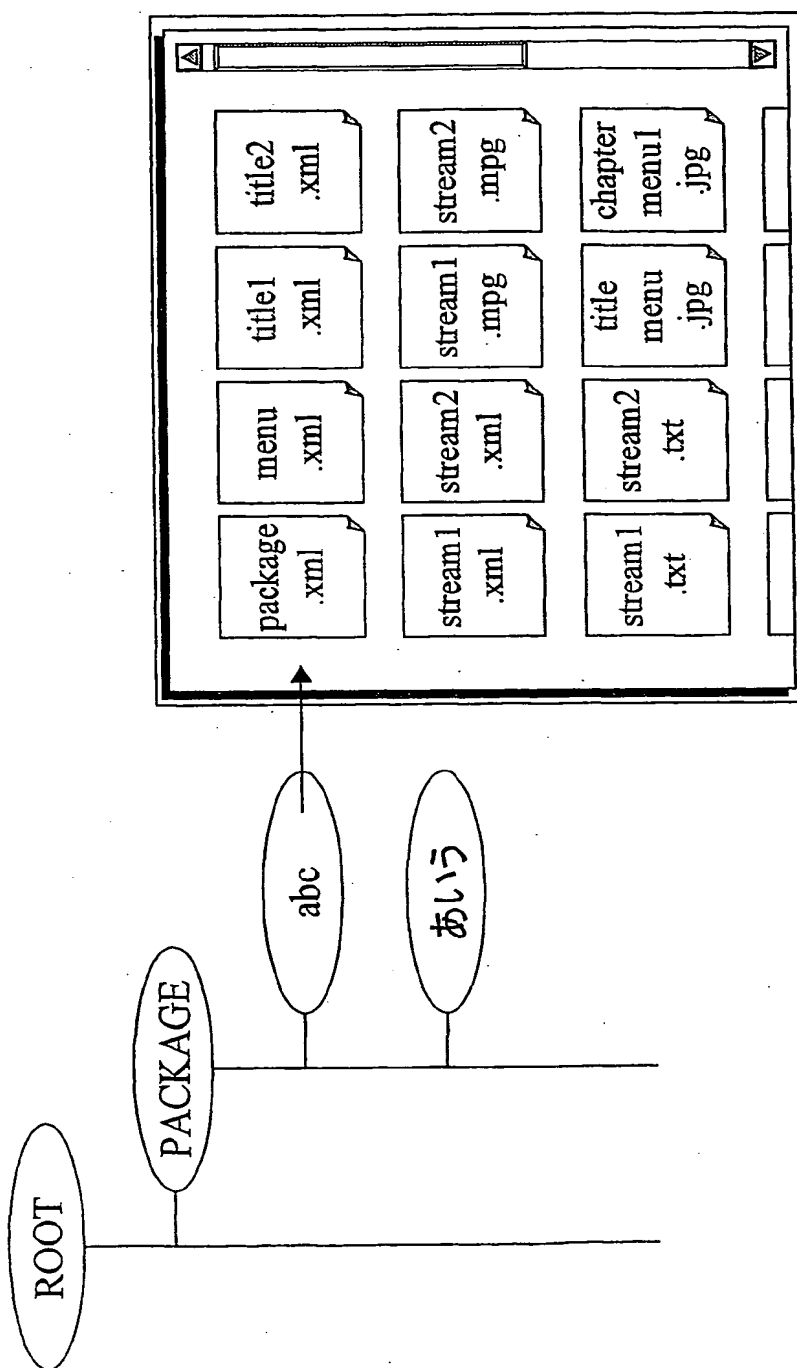


図15

16 / 39

図16

```
...
<PACKAGE>
  <GENERAL version="1.0"/>
  <ACCESS region="Japan" level="full" expire="2000.03.31"/>
  <UPDATE date="2000.2.15" auto="yes"
  <INTERNET URL="http://www.abc.co.jp">
  <MENU file="menu.xml">
  <TITLE_LIST>
    <TITLE number="1" file="title1.xml">
    ...
  </TITLE_LIST>
</PACKAGE>
...
```


17/39

図17

```
...  
<MENU>  
<MENU_PAGE page="1" image="titlenui.jpg">  
<TITLE column="1" row="1" title="1" object="button">劇場公開版</TITLE>  
<TITLE column="1" row="2" title="2" object="button">ディレクターズカット版  
</TITLE>  
</MENU_PAGE>  
<MENU>  
...
```

18 / 39

図 18

```

...
<TITLE titleno="1" level="full">
<LINK_LIST>
  <LINK ID="1" package="abc" title="1" chapter="1" time="00:00:00:00"/>
</LINK_LIST>
<CHAPTER_LIST>
<CHAPTER in="00:00:00:00" out="00:24:15:30"
  video="stream1.mpg" subtitle="stream1.txt">
<TIMELINE>
  <BRANCH level="full" message="過去の名作" id="1" in="00:01:00:00"
    out="00:11:00:00" jump="title2" />
  ...
</TIMELINE>
  ...
</CHAPTER>
</CHAPTER_LIST>
<INTERFACE>
  <PLAY level="free">
  <SETRATE level="full">
  ...
</INTERFACE>
</TITLE>
...

```

19 / 39

図19

```
...
<STREAM file="stream1.mpg">
  <ATTRIBUTE>
    <VIDEO coding="MPEG2" resolution="720x480" aspect="16:9"/>
    <AUDIO coding="AC-3" bitrate="256kbps" channel="5.1" language="English"/>
    <AUDIO coding="MPEG" bitrate="224kbps" channel="2" language="Japanese"/>
  </ATTRIBUTE>
  <TIMEMAP>
    <ENTRY duration="00:00:00:12" size="152000"/>
    <ENTRY duration="00:00:00:15" size="213000"/>
  ...
  </TIMEMAP>
</STREAM>
...
```

20 / 39

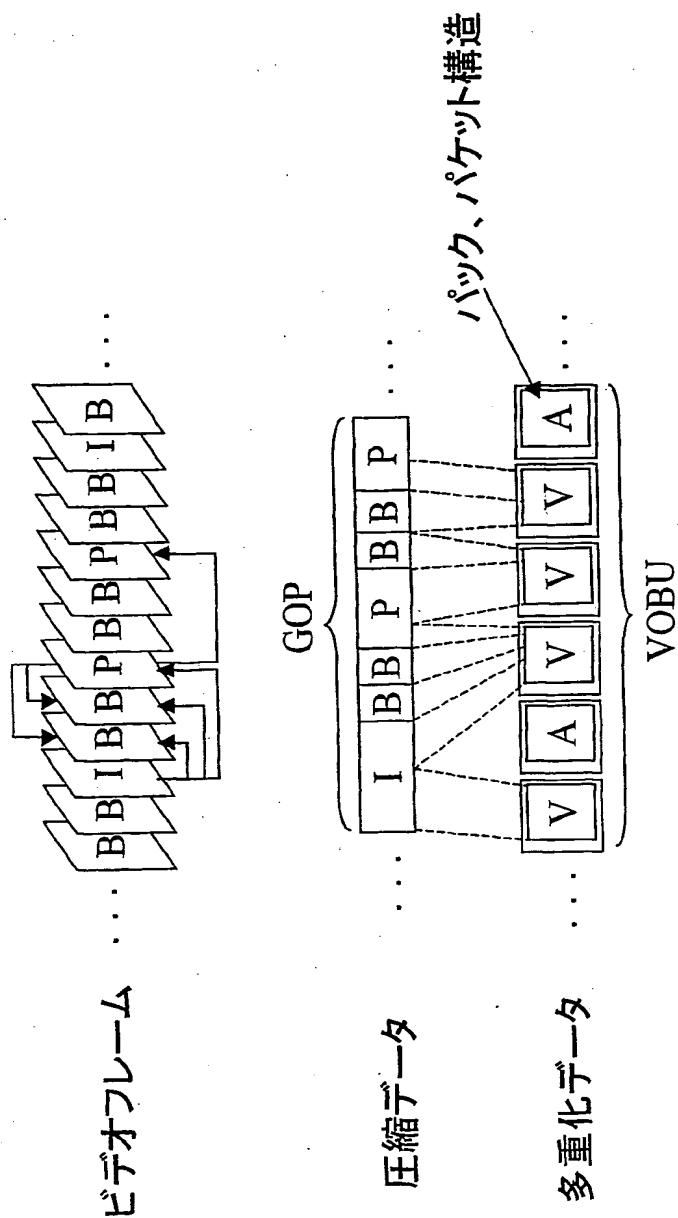
図20

```

...
<SUBTITLE>
<LANGUAGE language="Japanese" character="Shift-JIS" font="Gothic"
color="white" italic="off" bold="off" underline="off">
<TEXT in="00:00:05:32" out="00:00:05:42">元氣だったか？</TEXT>
<TEXT in="00:00:05:45" out="00:00:05:50" italic="on">ああ、</TEXT>
<TEXT in="00:00:06:02" out="00:00:06:12" color="blue">なにかあったのか？
</TEXT>
<TEXT in="00:00:06:17" out="00:00:06:25">じつは...</TEXT>
...
</LANGUAGE>
...
</SUBTITLE>
...
```

21 / 39

図21



22 / 39

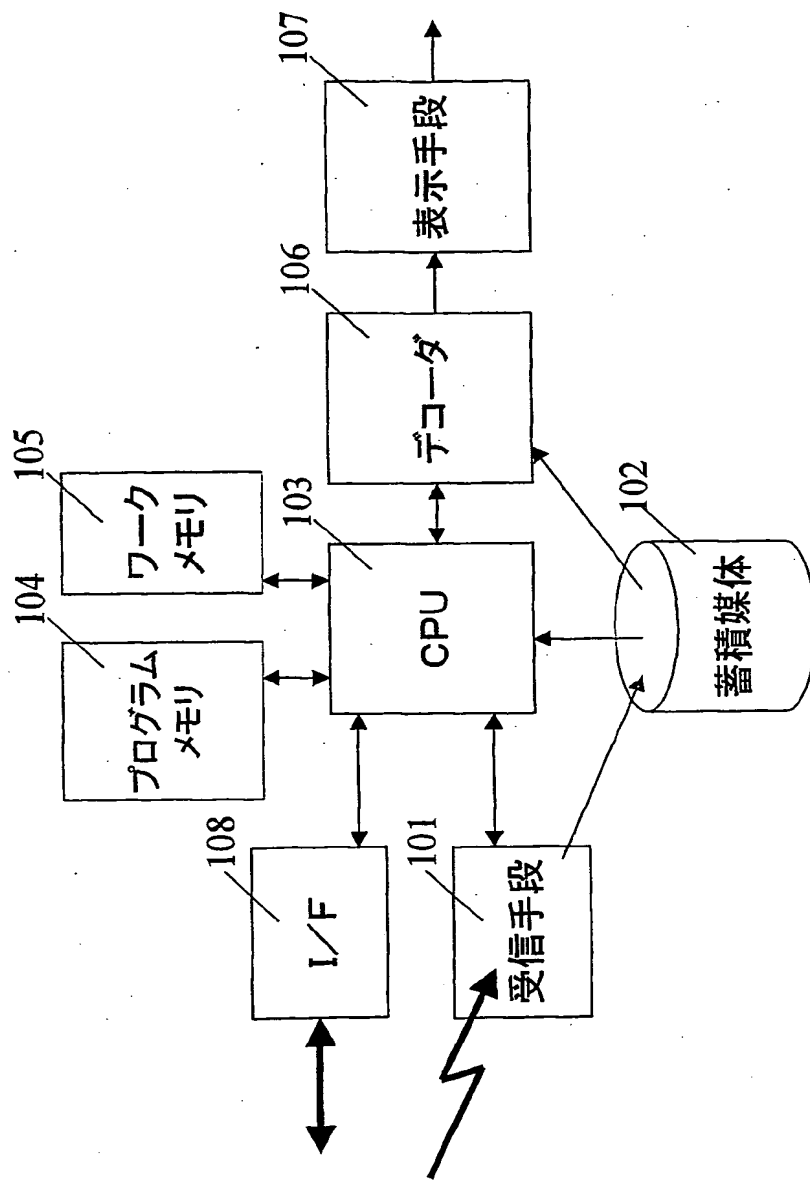
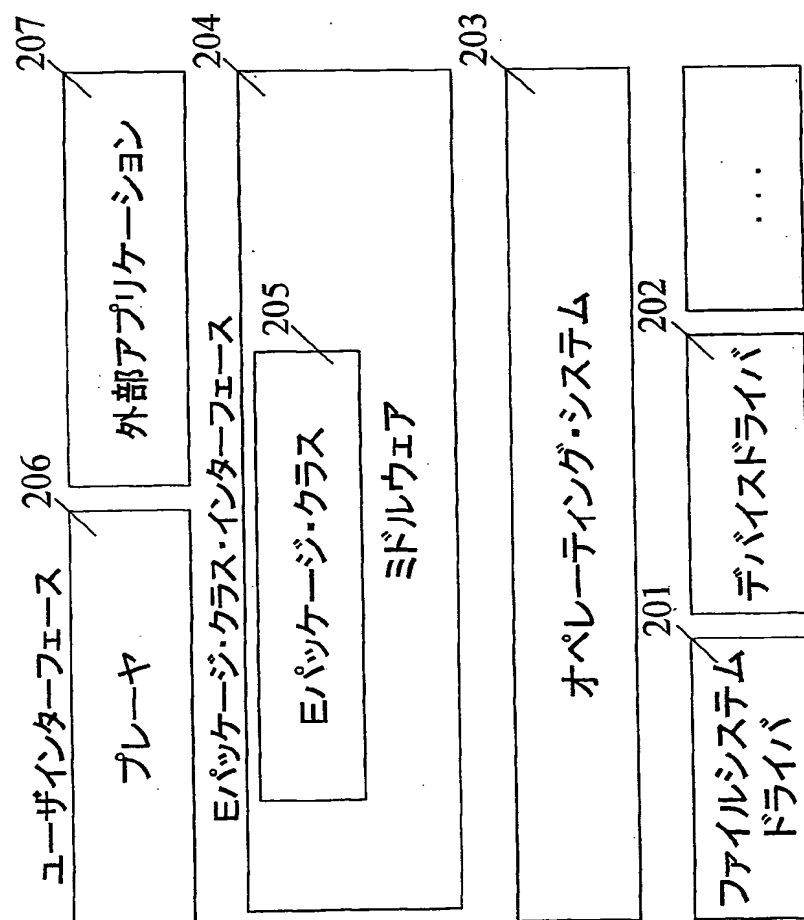


図22

23 / 39

図23



24 / 39

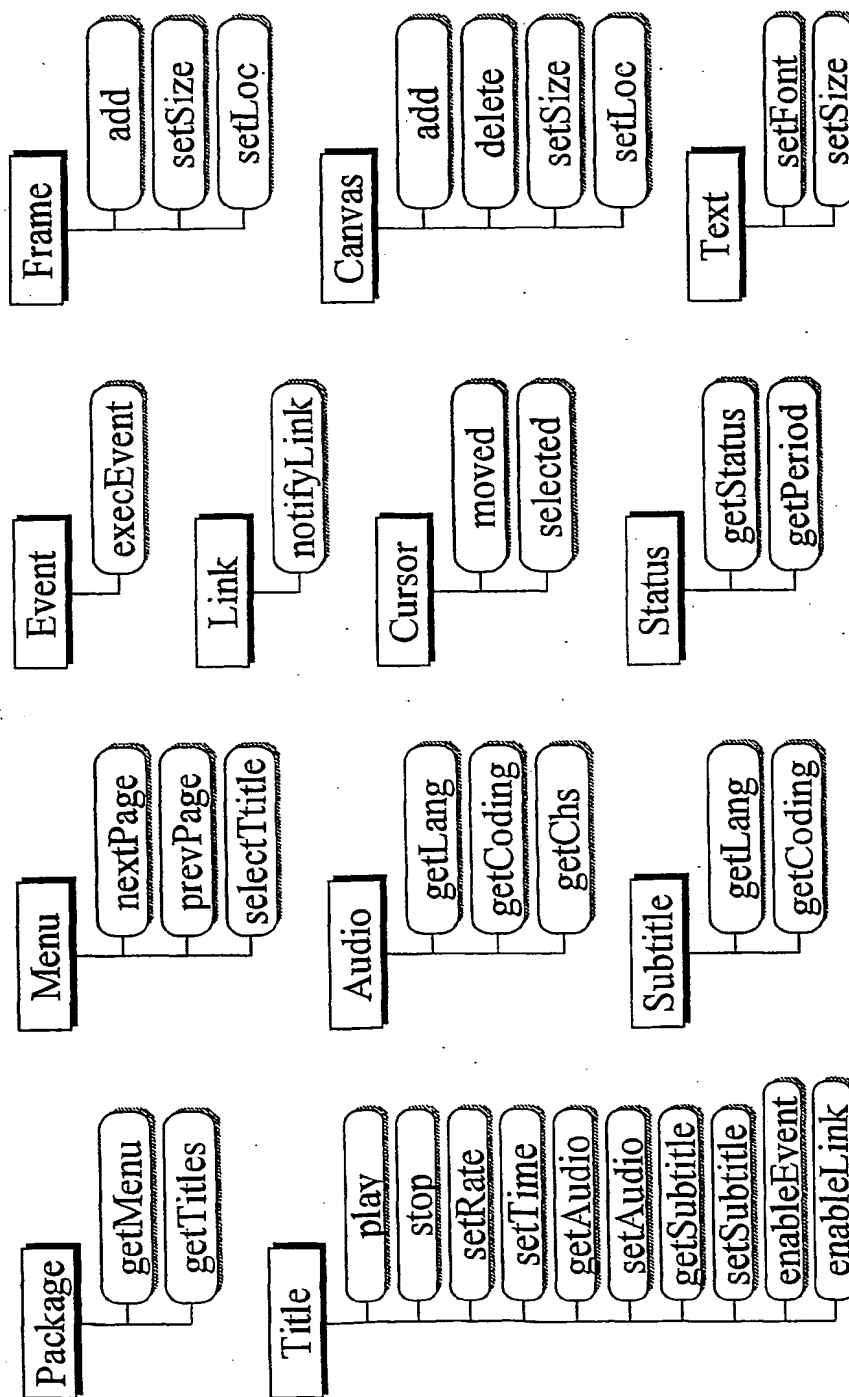


图24

25 / 39

図25

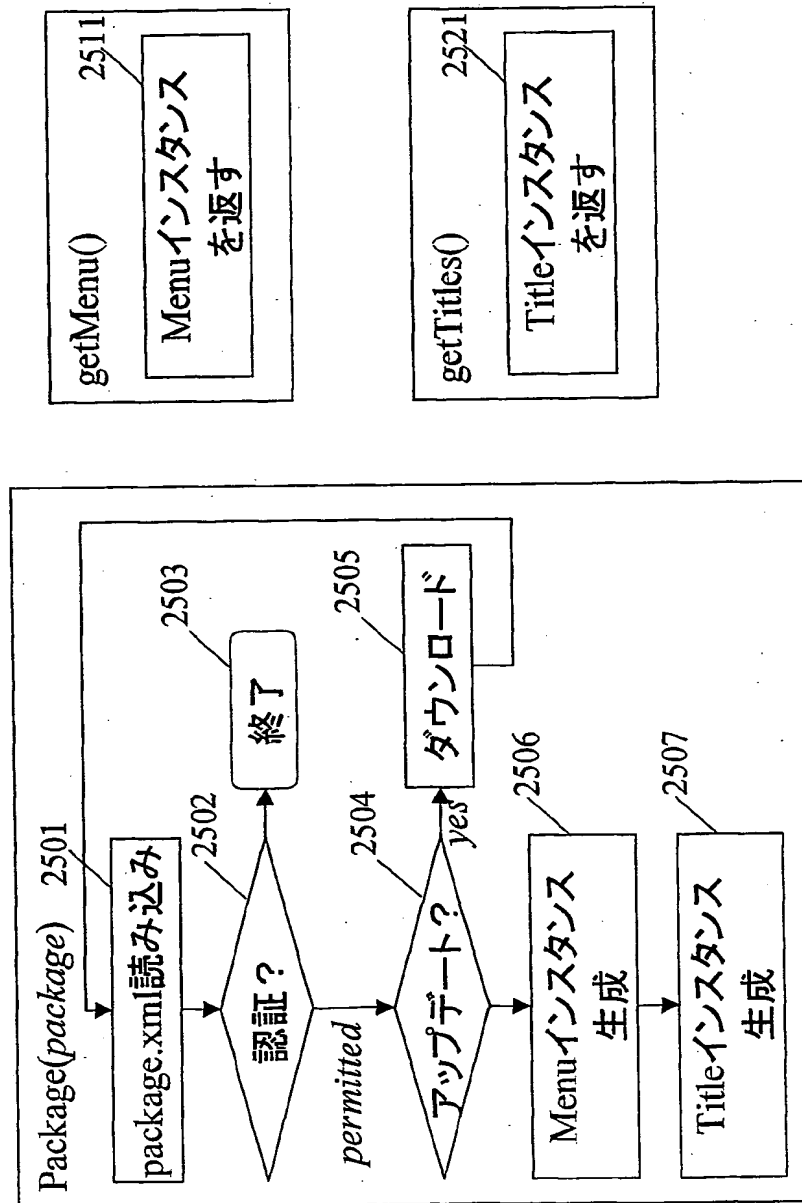
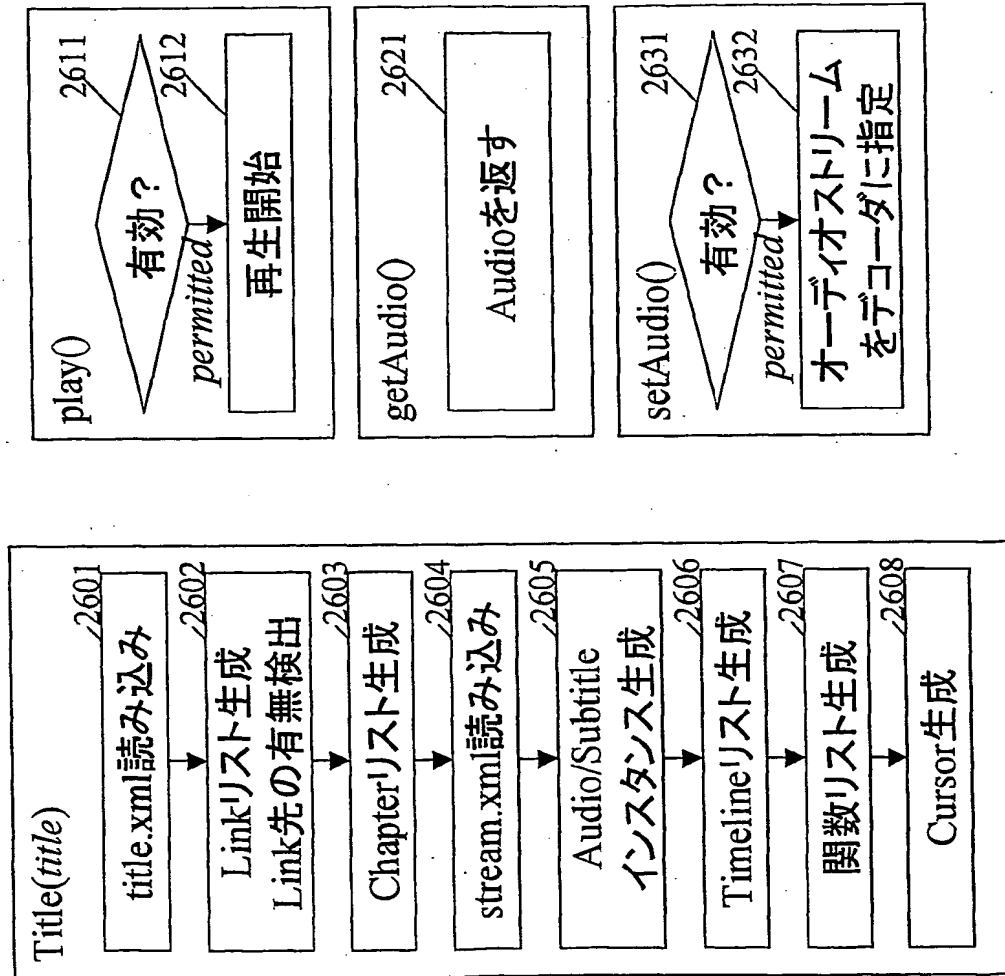
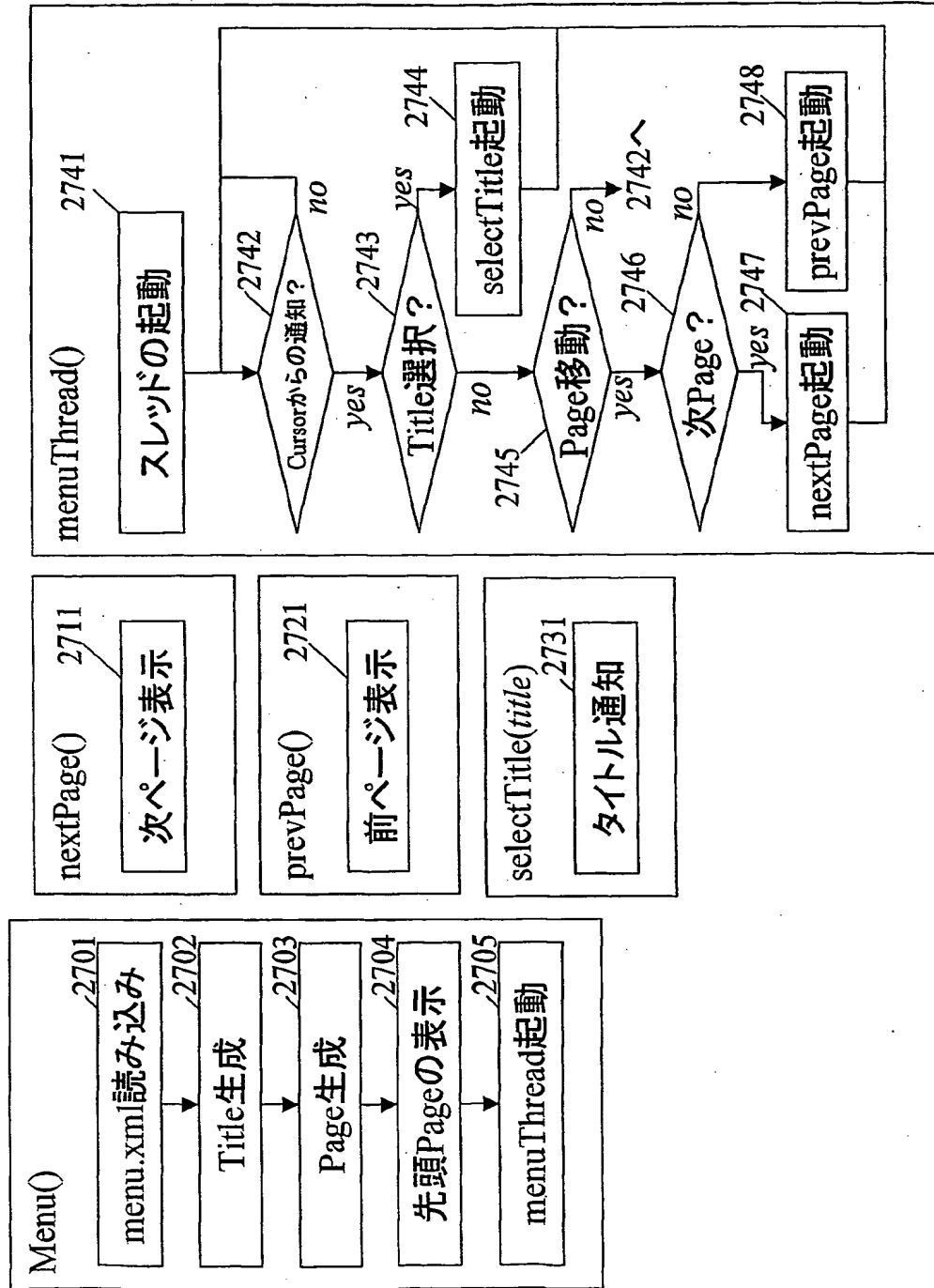


図 26



27 / 39

図27



30 / 39

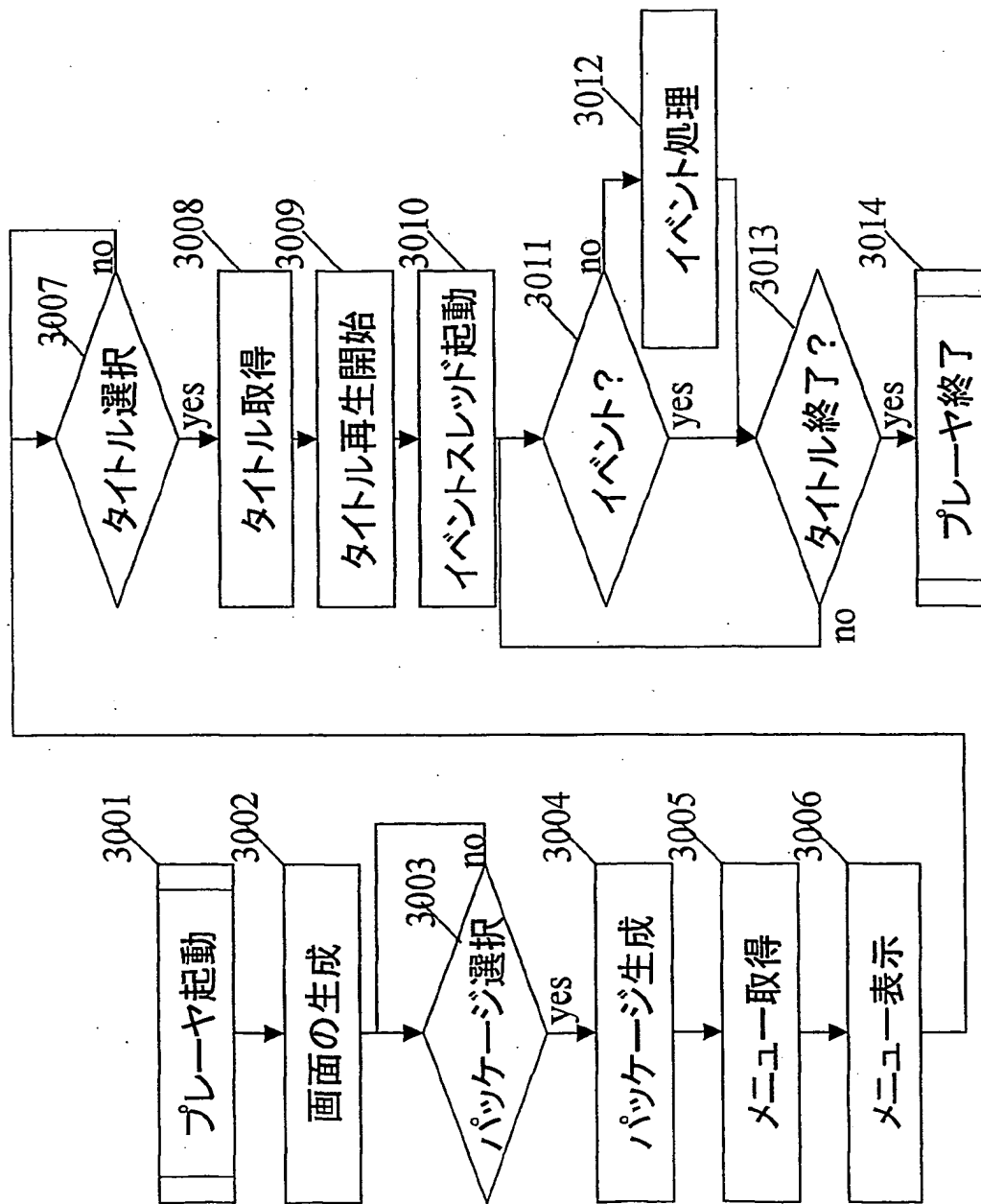
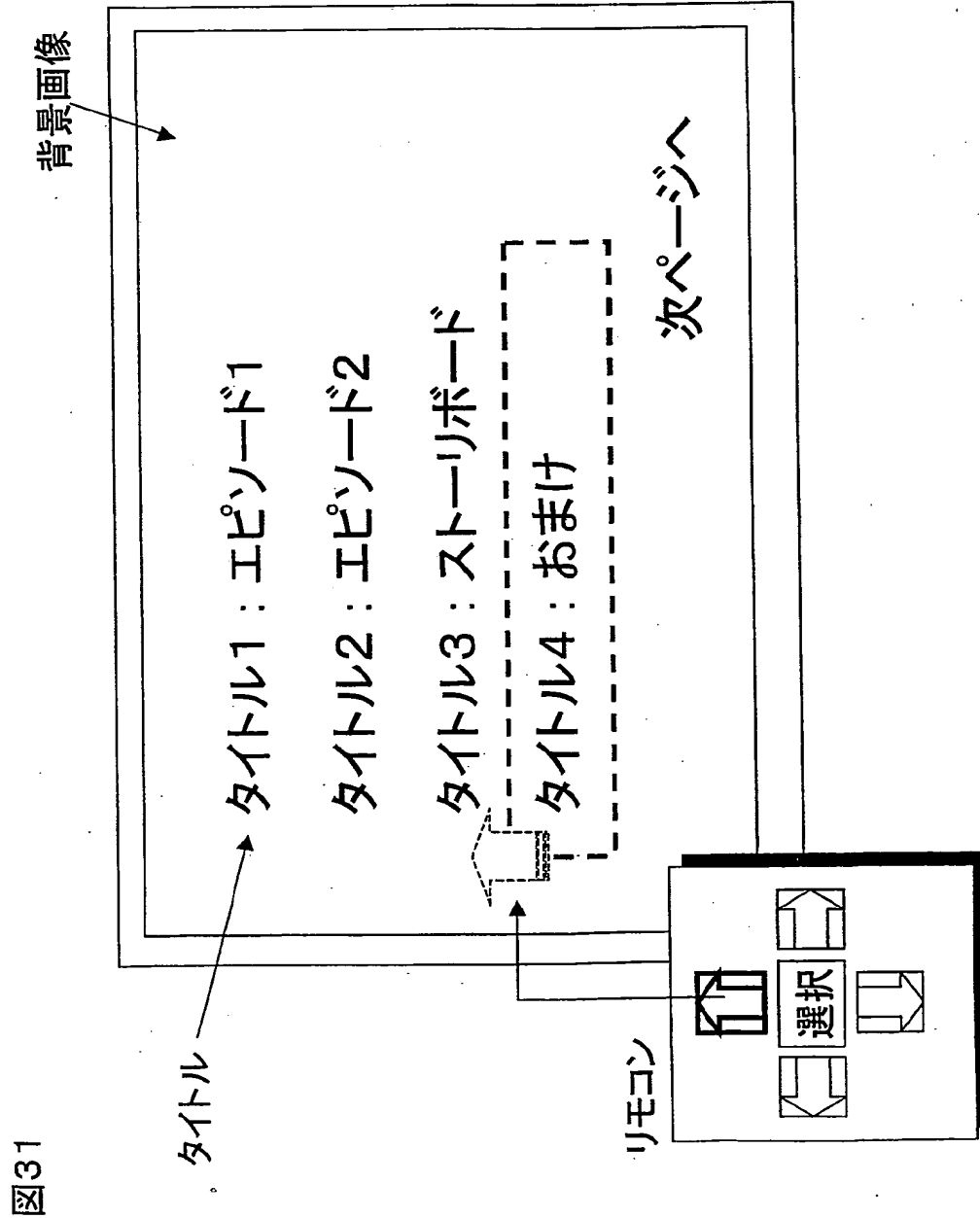


図30

31 / 39



32 / 39

図32

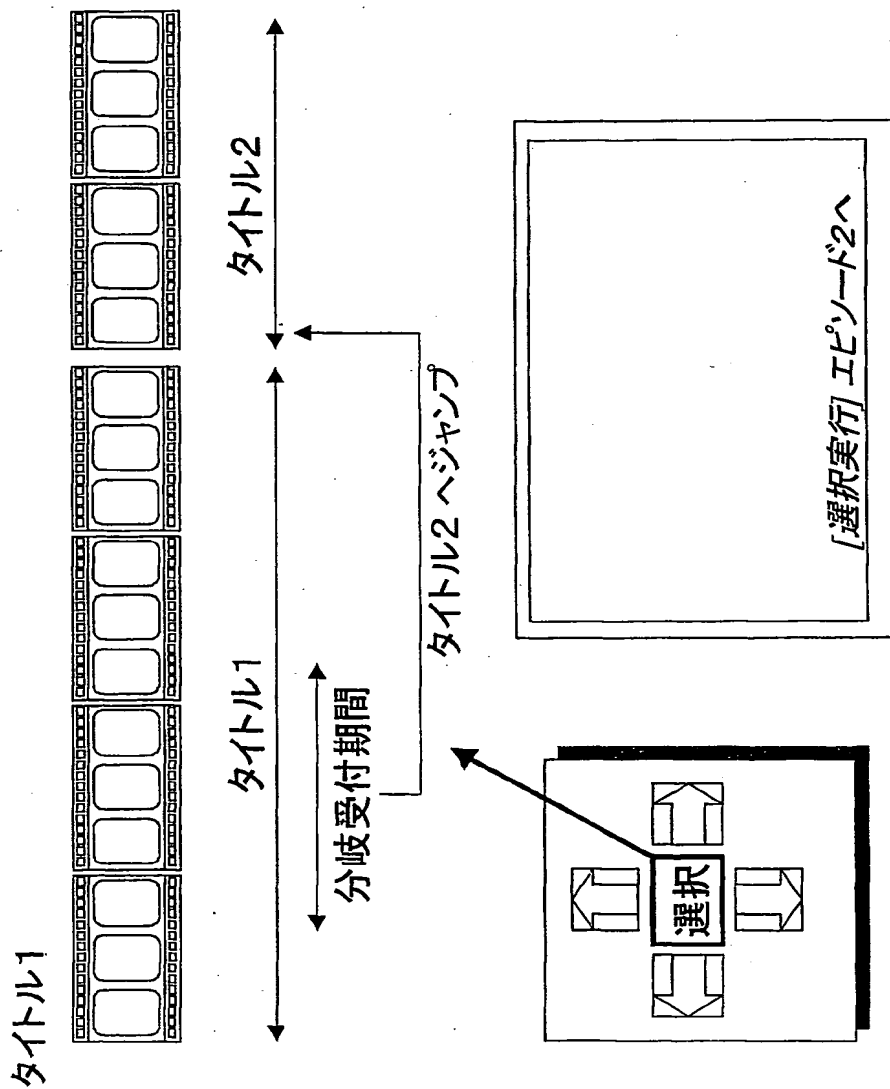


図33

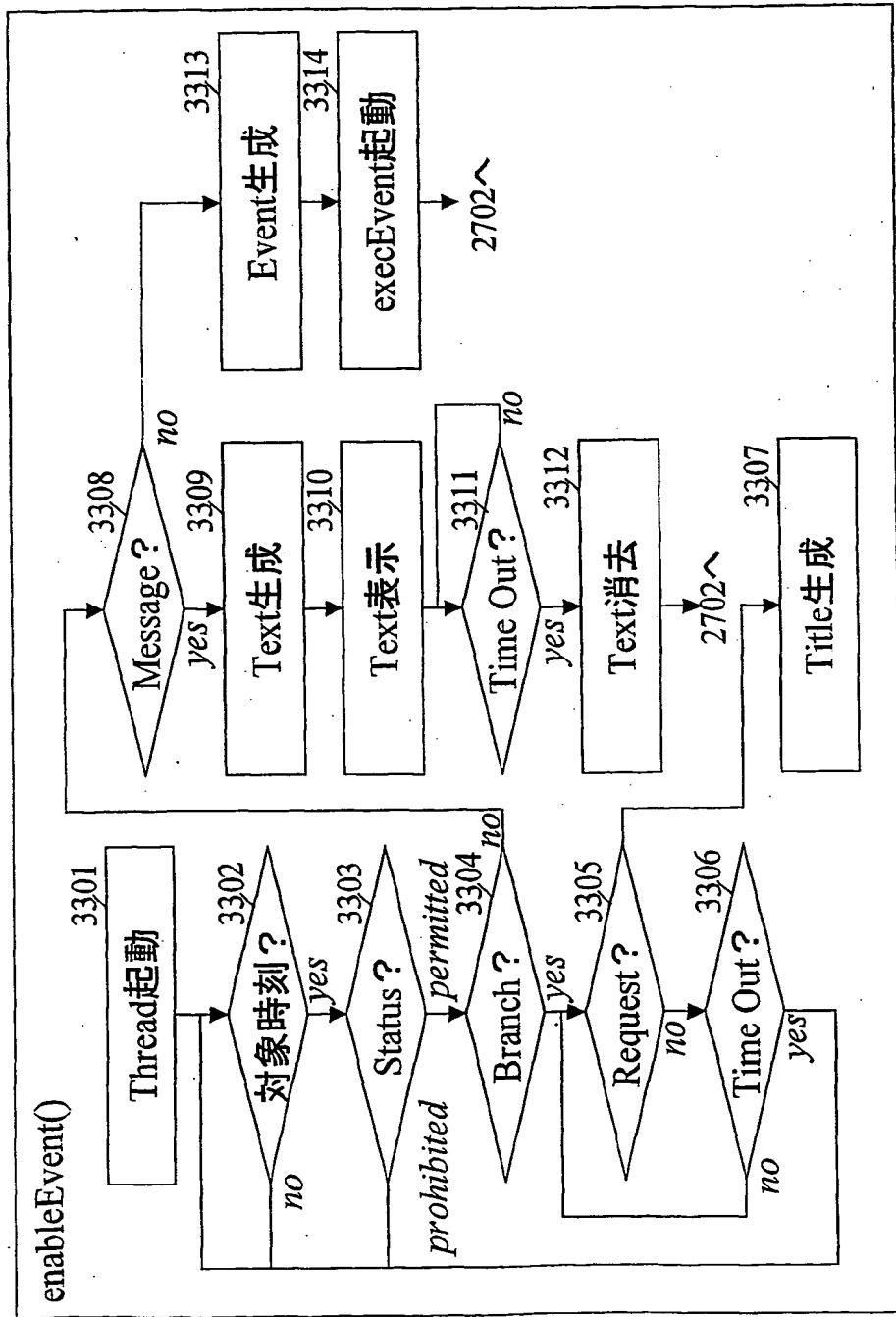


図34

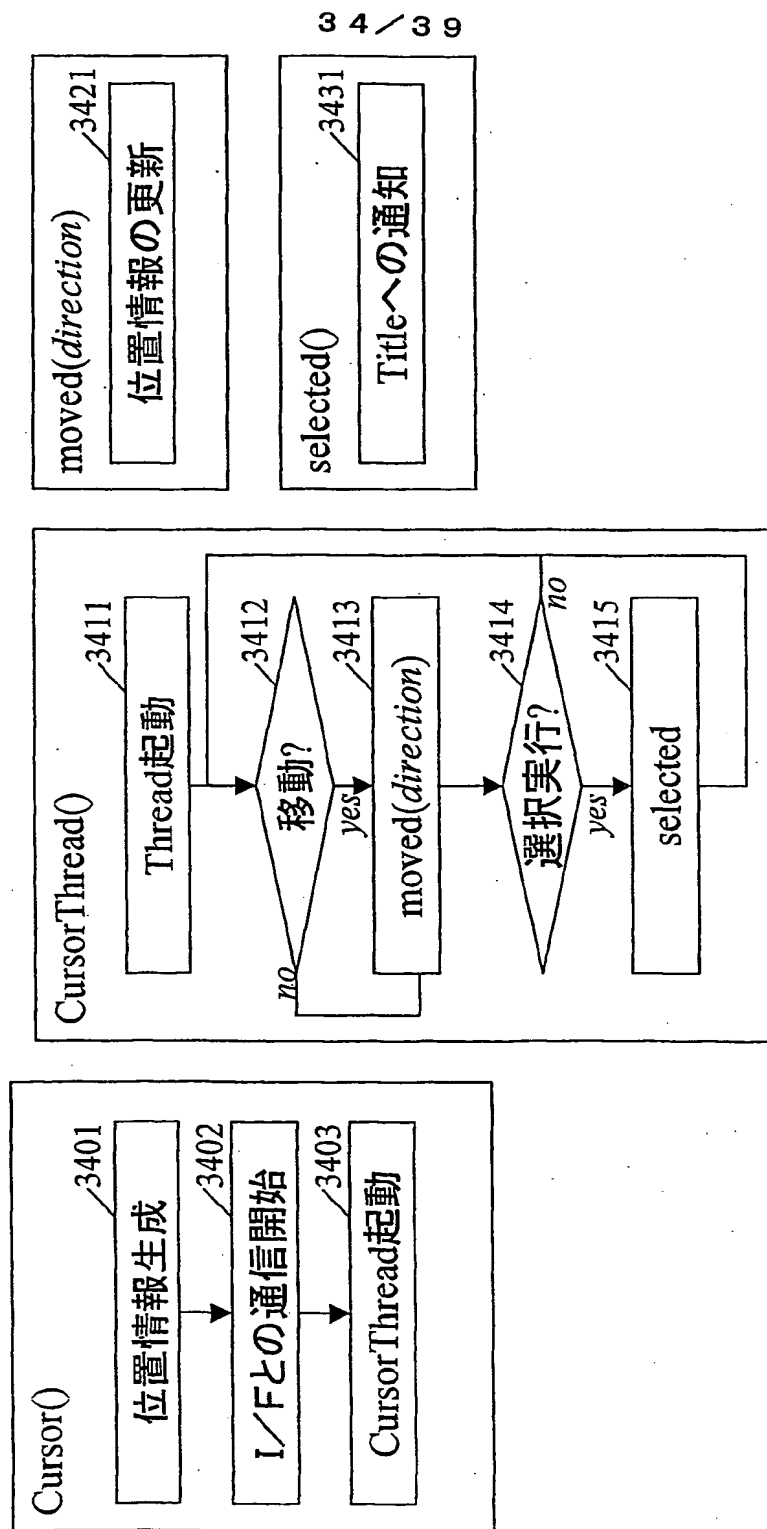


図35

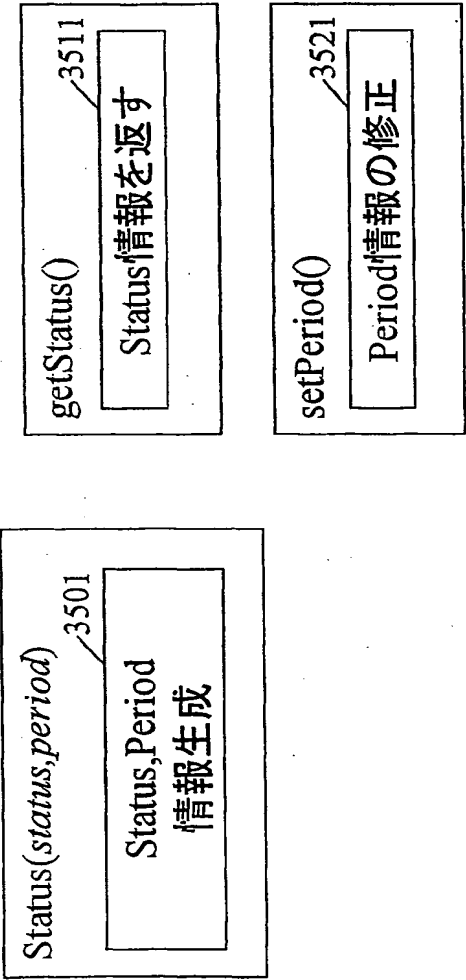
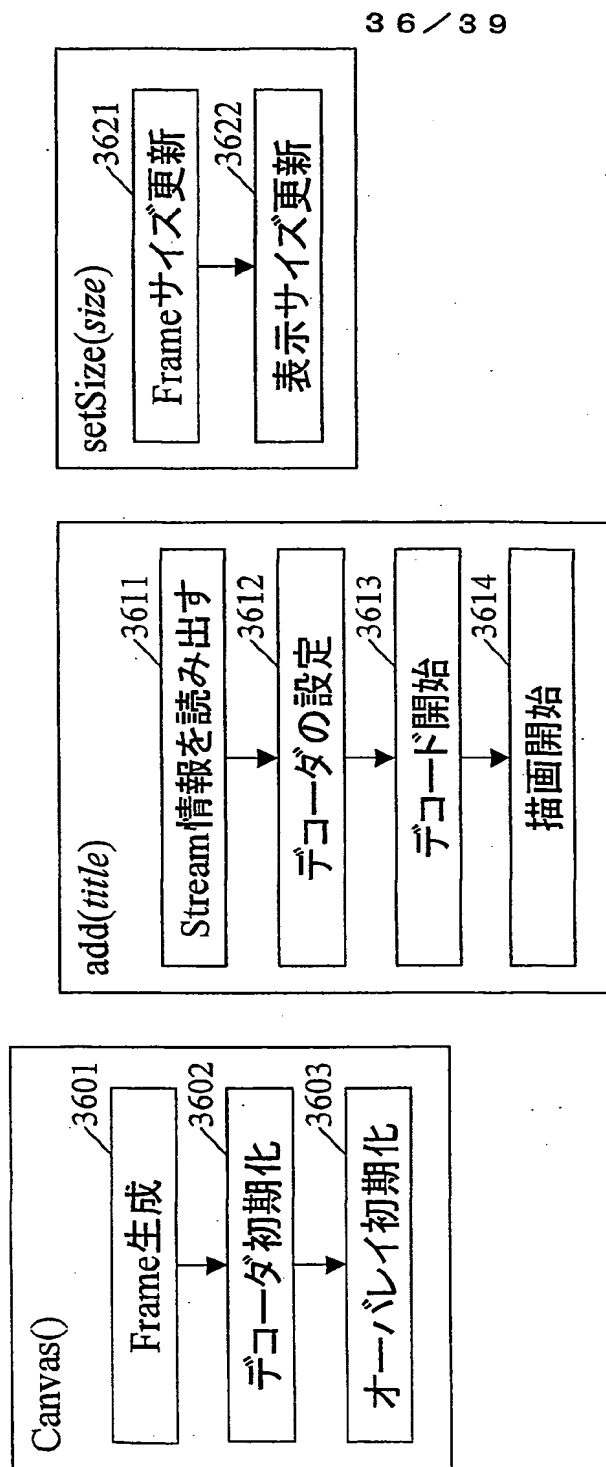


図36



37 / 39

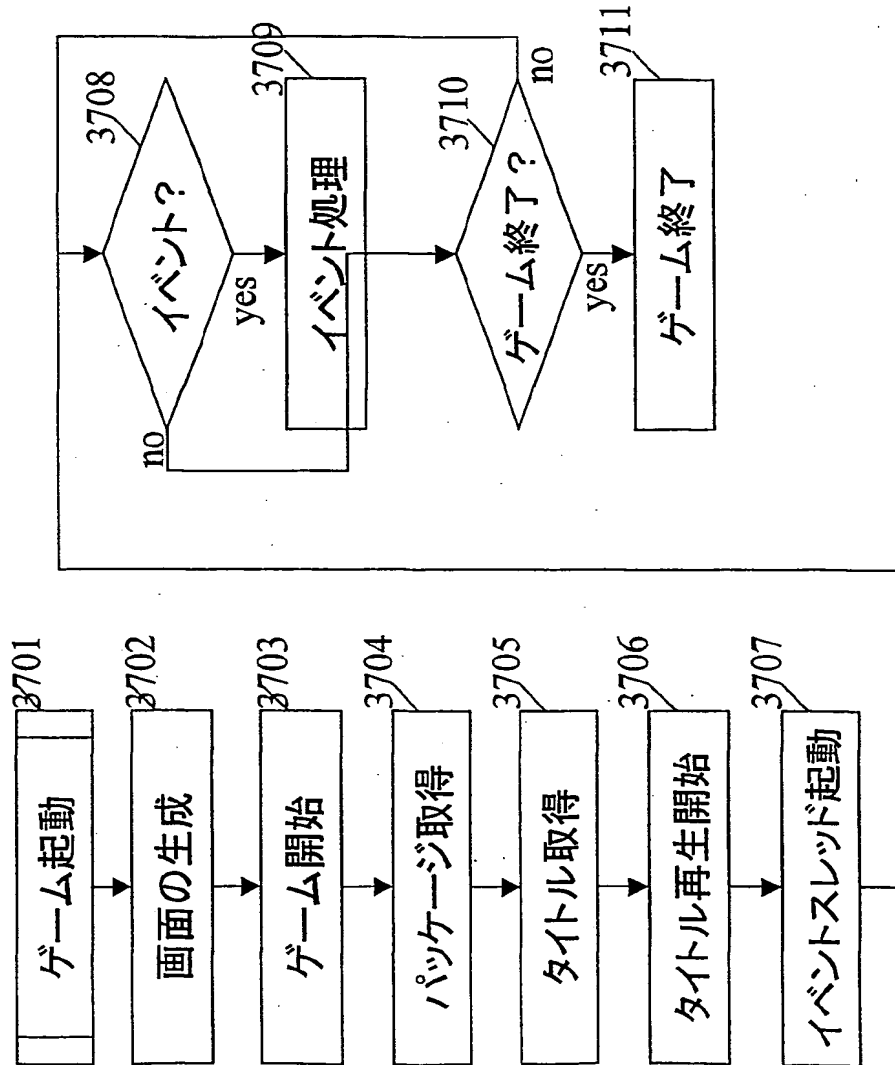


図37

38 / 39

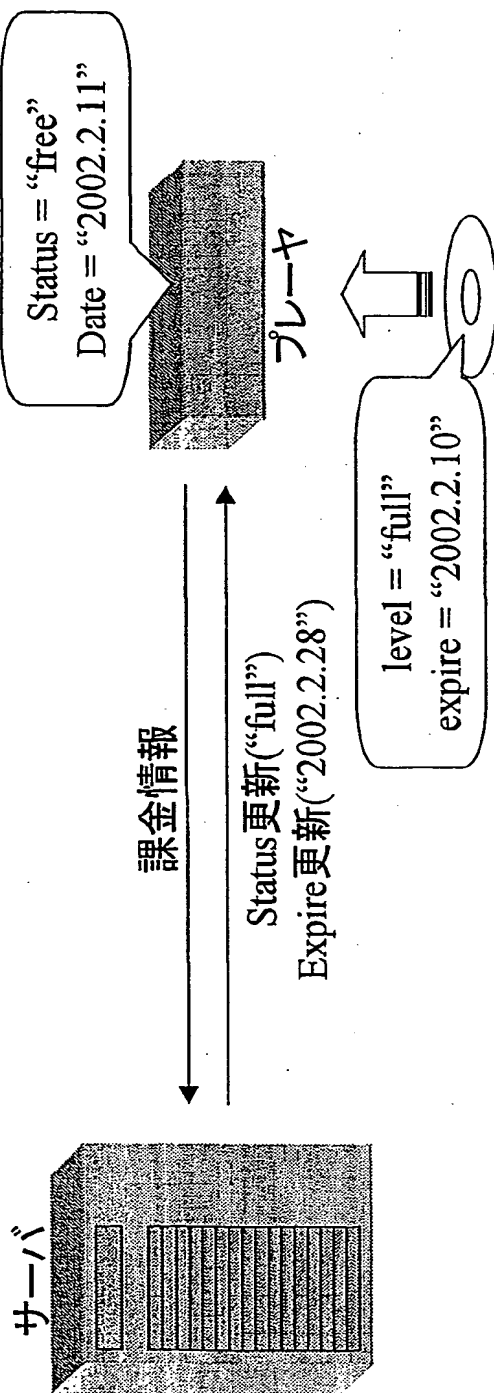
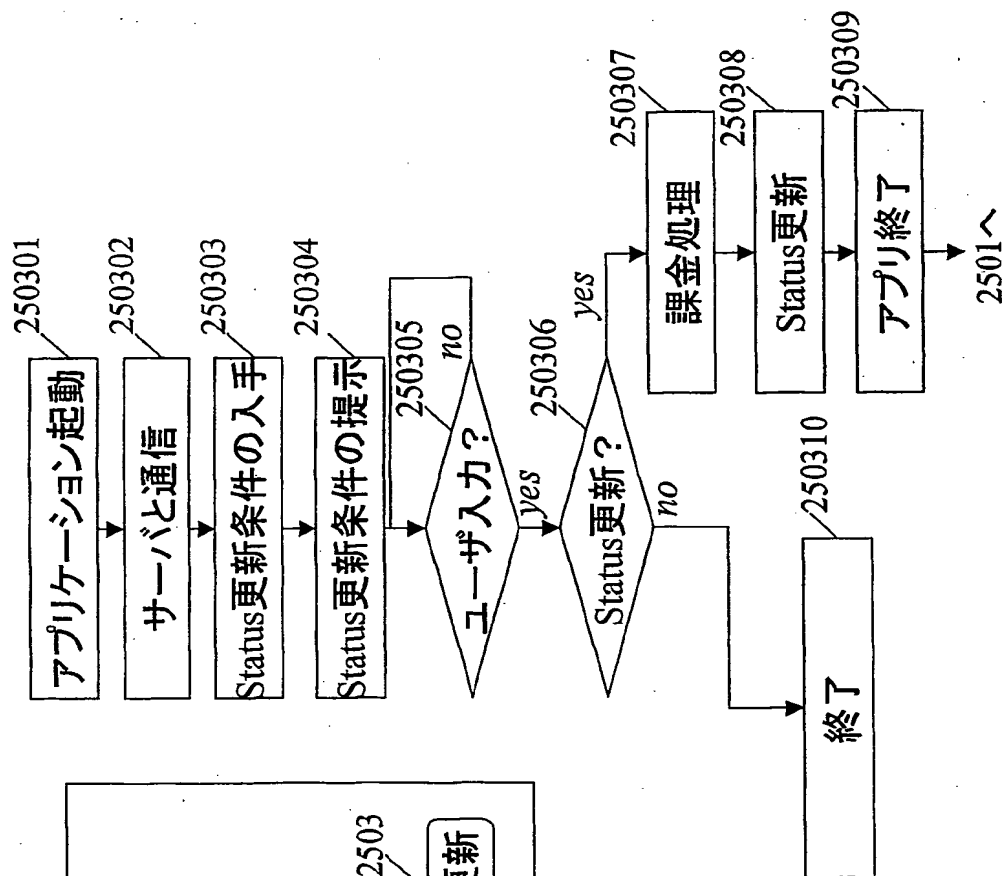
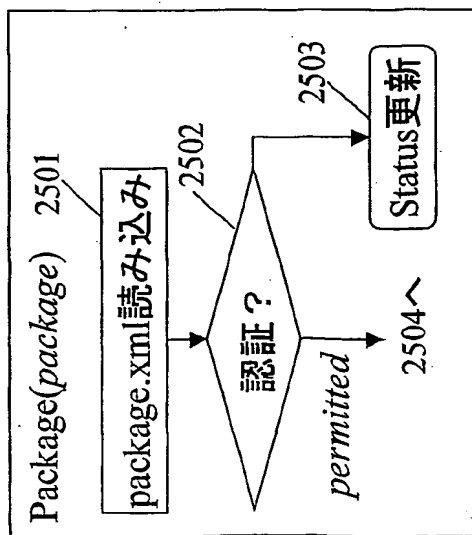


図38

図39



INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP02/03152

A. CLASSIFICATION OF SUBJECT MATTER

Int.Cl⁷ H04N5/91, H04N5/92, H04N5/76, G11B20/10, G11B27/00

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

Int.Cl⁷ H04N5/76-956, G11B20/10, G11B27/00, G06F9/06, G06F9/445

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Jitsuyo Shinan Koho	1922-1996	Toroku Jitsuyo Shinan Koho	1994-2002
Kokai Jitsuyo Shinan Koho	1971-2002	Jitsuyo Shinan Toroku Koho	1996-2002

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	JP 2000-113066 A (Fujitsu Ltd., Hitachi Software Engineering Co., Ltd.), 21 April, 2000 (21.04.00), Full text; Figs. 1 to 45 (Family: none)	1-6
Y	JP 2000-057746 A (Toshiba Corp., Toshiba Digital Media Engineering Kabushiki Kaisha), 25 February, 2000 (25.02.00), Full text (particularly, the passage explaining applied examples in paragraph [0327] and later paragraphs); Figs. 1 to 29 (Family: none)	1-6
Y	JP 8-279963 A (Sony Corp.), 22 October, 1996 (22.10.96), Full text; Figs. 1 to 9 & US 6046780 A	1-6

☒ Further documents are listed in the continuation of Box C.
 ☐ See patent family annex.

* Special categories of cited documents:	"I" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A" document defining the general state of the art which is not considered to be of particular relevance	"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E" earlier document but published on or after the international filing date	"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&" document member of the same patent family
"O" document referring to an oral disclosure, use, exhibition or other means	
"P" document published prior to the international filing date but later than the priority date claimed	

 Date of the actual completion of the international search
27 June, 2002 (27.06.02)

 Date of mailing of the international search report
09 July, 2002 (09.07.02)

 Name and mailing address of the ISA/
Japanese Patent Office

Authorized officer

Facsimile No.

Telephone No.

INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP02/03152

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	JP 10-150639 A (Hitachi, Ltd.), 02 June, 1998 (02.06.98), Full text; Figs. 1 to 4 (Family: none)	1-6
Y	JP 11-112934 A (Victor Company Of Japan, Ltd.), 23 April, 1999 (23.04.99), Full text; Figs. 1 to 12 (Family: none)	1-6

国際調査報告

国際出願番号 PCT/JPO2/03152

A. 発明の属する分野の分類 (国際特許分類 (IPC))

Int. Cl¹ H04N5/91, H04N5/92, H04N5/76, G11B20/10, G11B27/00

B. 調査を行った分野

調査を行った最小限資料 (国際特許分類 (IPC))

Int. Cl¹ H04N5/76-956, G11B20/10, G11B27/00, G06F9/06, G06F9/445

最小限資料以外の資料で調査を行った分野に含まれるもの

日本国実用新案公報 1922-1996年
 日本国公開実用新案公報 1971-2002年
 日本国登録実用新案公報 1994-2002年
 日本国実用新案登録公報 1996-2002年

国際調査で使用した電子データベース (データベースの名称、調査に使用した用語)

C. 関連すると認められる文献

引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号
Y	JP 2000-113066 A (富士通株式会社、日立ソフトウェアエンジニアリング株式会社) 2000.04.21 全文、図1-45 (ファミリーなし)	1-6
Y	JP 2000-057746 A (株式会社東芝、東芝デジタルメディアエンジニアリング株式会社) 2000.02.25 全文 (特に段落【0327】以降の応用例の説明箇所)、図1-29 (ファミリーなし)	1-6
Y	JP 8-279963 A (ソニー株式会社) 1996.10.22 全文、図1-9 & US 6046780 A	1-6

☒ C欄の続きにも文献が列挙されている。☐ パテントファミリーに関する別紙を参照。

* 引用文献のカテゴリー

「A」 特に関連のある文献ではなく、一般的技術水準を示すもの

「E」 国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの

「L」 優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献 (理由を付す)

「O」 口頭による開示、使用、展示等に言及する文献

「P」 国際出願日前で、かつ優先権の主張の基礎となる出願

の日の後に公表された文献

「T」 国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの

「X」 特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの

「Y」 特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの

「&」 同一パテントファミリー文献

国際調査を完了した日

27.06.02

国際調査報告の発送日

09.07.02

国際調査機関の名称及びあて先

日本国特許庁 (ISA/JIP)

郵便番号100-8915

東京都千代田区霞が関三丁目4番3号

特許庁審査官 (権限のある職員)

石丸 昌平

5C

9559

電話番号 03-3581-1101 内線 6977

C (続き) . 関連すると認められる文献		
引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号
Y	JP 10-150639 A (株式会社日立製作所) 1998.06.02 全文、図1-4 (ファミリーなし)	1-6
Y	JP 11-112934 A (日本ビクター株式会社) 1999.04.23 全文、図1-12 (ファミリーなし)	1-6

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record.**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☒ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☒ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☒ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☒ **OTHER: _____**

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.